

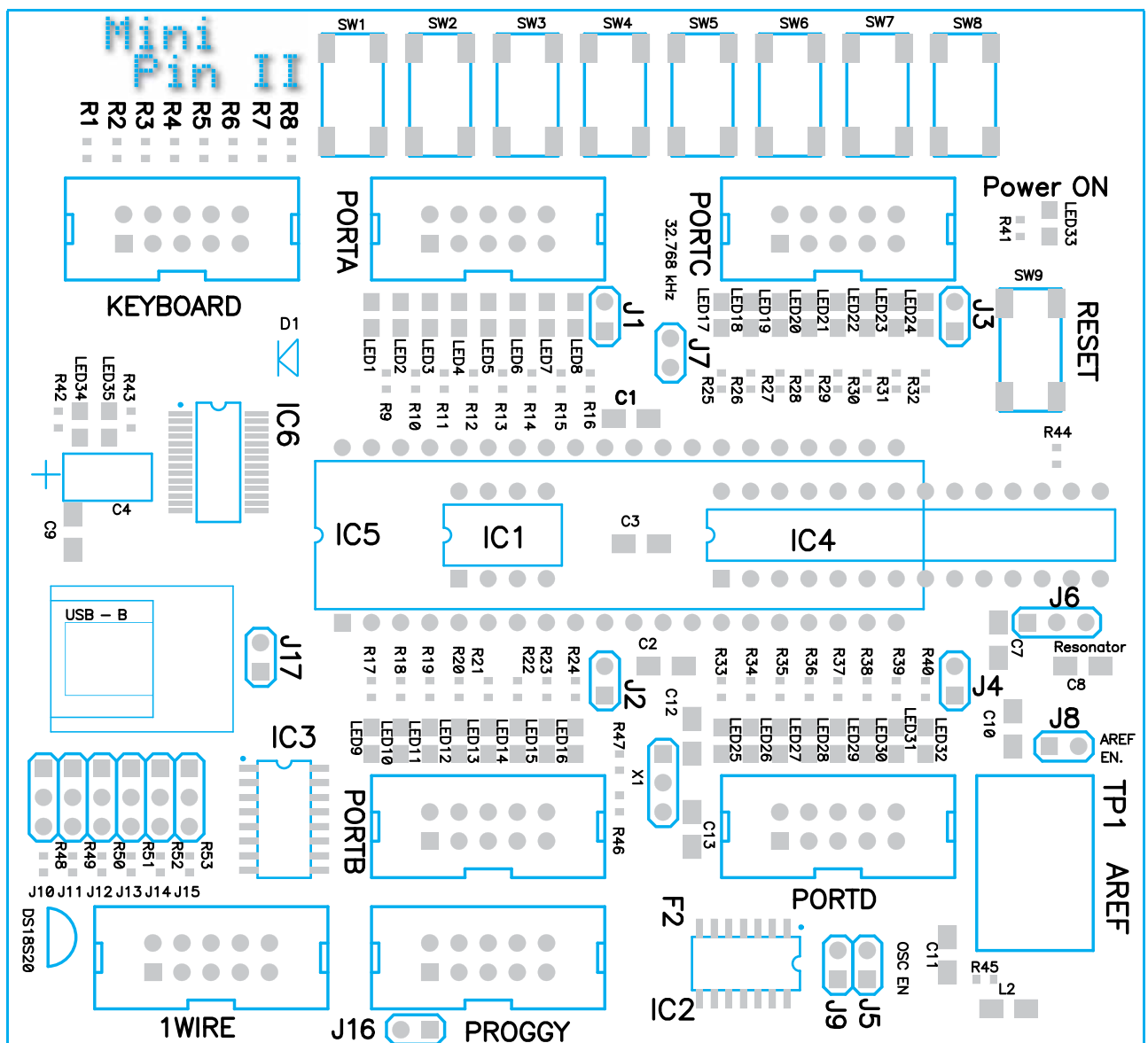
MiniPin II development board

By JURIJ MIKELN

MiniPin II is a smaller brother of MegaPin. Smaller, but still very user friendly and versatile development board, where programs for AVR microcontrollers may be tested and evaluated.

Key MiniPin II features:

1. supports all AVR microcontrollers in 8-pin, 28-pin & 40 pin DIL packages (except ATmega8515, 8535, i.e. older AVR's),
2. 6 built-in sockets for 1-Wire DS18S20 temperature sensors, which can be connected to any I/O Port,
3. 8 LEDs on each of four I/O ports (PortA, PortB, PortC & PortD),
4. IDC I/O connectors on four ports: (PortA, PortB, PortC, PortD),
5. built-in 250 mA fuse to protect USB from short circuit on board,
6. built-in USB communication to UART (not on MiniPin II B version!),
7. built-in external oscillator circuit to clock microcontroller from an external quartz crystal,
8. built-in 3-pin socket for ceramic resonator,
9. built-in 2-pin socket for 32,768 kHz quartz crystal,
10. built-in adjustable reference voltage for the AVR's ADC,
11. built-in keyswitch array which can be connected to any I/O Port,
12. built-in Power-ON LED,
13. built-in serial communication Tx/Rx LEDs.



Connecting your programmer to the MiniPin II board.

When connecting your programmer to the MiniPin II board, be sure that you connect the programmer's socket to the socket labelled "PROGGY", as shown in Figure 1. If you don't own a Proggy programmer, make sure that you connect its signals to the MiniPin II board, as follows:

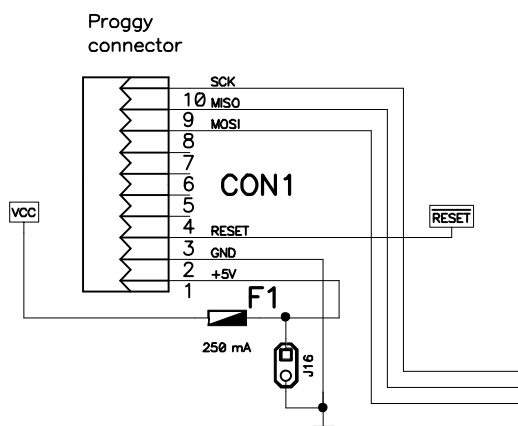


Figure 1: SPI Signals on Proggy IDC connector

The MiniPin II board is powered by the programmer and it has a 250 mA fuse in series with the Vcc power line, to protect your PC's USB port in case you have a short-circuit on your MiniPin II extension board.

Programming AVR's in MiniPin II

MiniPin II can program AVR microcontrollers that are mounted on the MiniPin II board itself - just mount your AVR chip in the appropriate socket, connect your programmer to MiniPin II and your AVR chip is ready for programming.

ISP programming from within AVR Studio 4

Mount your AVR microcontroller in the appropriate socket and click the AVR icon from within the AVR Studio 4 program (Figure 2).

Using the **Main** tab, select the microcontroller that you want to program (Figure 3).

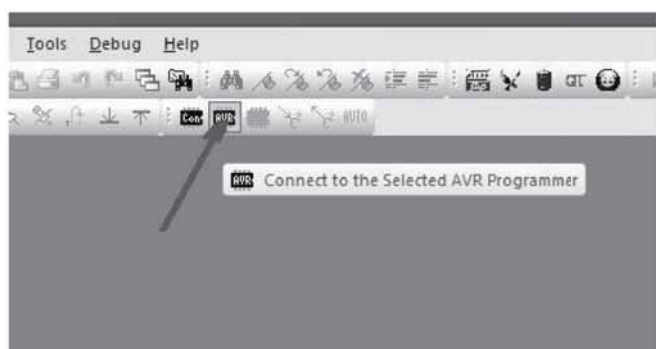


Figure 2: Connect to Programmer icon

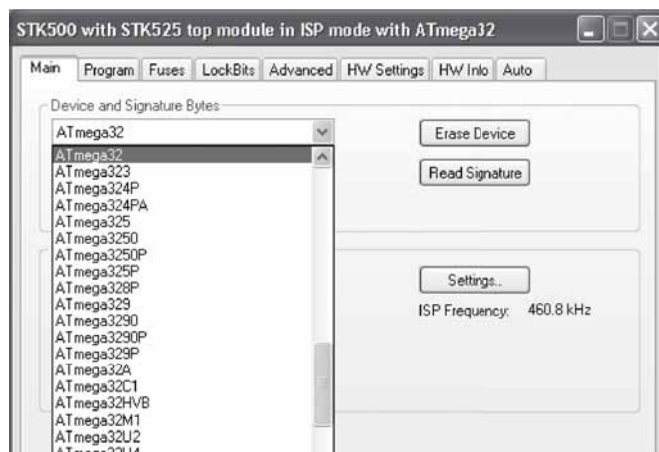


Figure 3: Select microcontroller

To insure that communication is OK, and that you have selected the correct microcontroller, press the **Read Signature** button. If the read signature corresponds to the microcontroller you are programming, then you may proceed with programming. If the signature doesn't match, then select the correct microcontroller from the list. If the program can't even read the microcontroller's signature, then there is something wrong with either the microcontroller, or some connections on the MiniPin II board are wrong (i.e. remove load from PortB).

ISP programming from within Bascom-AVR

Mount your AVR microcontroller in an appropriate socket, and within Bascom-AVR select **Options/Programmer**. A window like that shown in Figure 4 will pop-up.

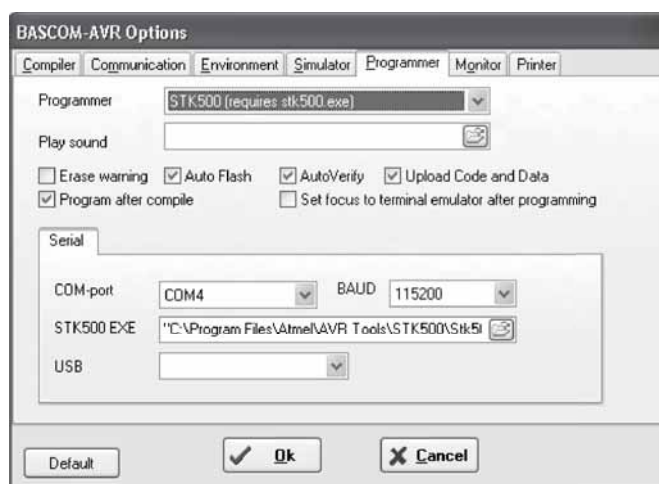


Figure 4: Bascom programming setting

In the **Programmer** window, select STK500 (requires stk500.exe), and set **COM-port** to the same COM port number as seen in the Device Manager program (see the full description regarding this subject in the MegaPin chapter). Then double-click the **folder** icon (to the right of the STK500 EXE window) and select the path to your STK500.exe file (which is usually the same as that shown

in Figure 4). Set the Baud rate to 57,600. Click OK and your Bascom-AVR settings are all done.

ISP programming in Bascom-AVR – the STK500 Native driver

Figure 5 shows the proper settings for programming using the **STK500 native driver**.

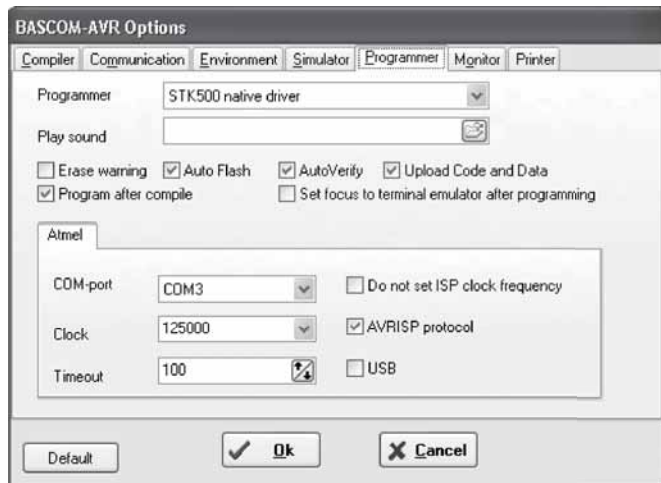


Figure 5: STK500 native programmer settings

1-Wire® sensors

The MiniPin II board has circuitry built-in to allow the use of popular DS18S20 temperature sensors. Please note that three sensors (J10-J12) are individually connected to their own I/O pin while those connected to J13 – J15 share the same I/O pin. For easier programming, sockets are labelled as follows: J10 connects to PortX.0, J11 connects to PortX.1 etc. All data lines are pulled up to Vcc by 4k7 resistors. Connect CON5 to any I/O port (PortA, PortB, PortC, PortD) with a flat cable and you can now measure temperature with your sensor(s). For a Bascom sample program, please refer to "Bascom-AVR sample programs for use with MiniPin II" at the end of this chapter.

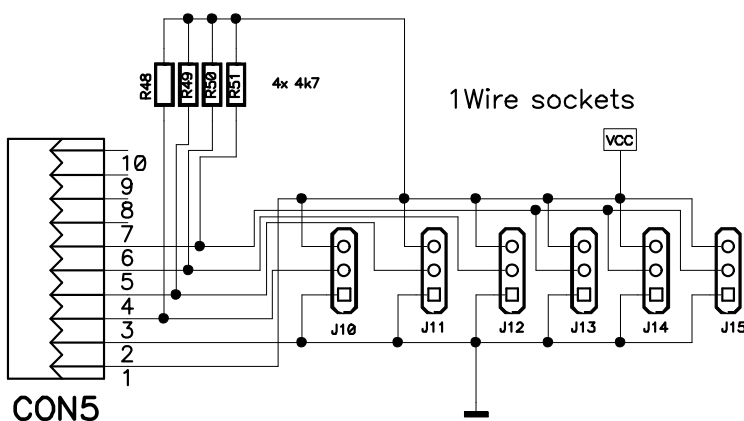


Figure 6: 1-Wire connections

Keyboard

The MiniPin II board has 8 keyswitches built-in, with protection resistors and a connector which allows the user to connect the keyswitches to any free I/O Port connector on MiniPin II.

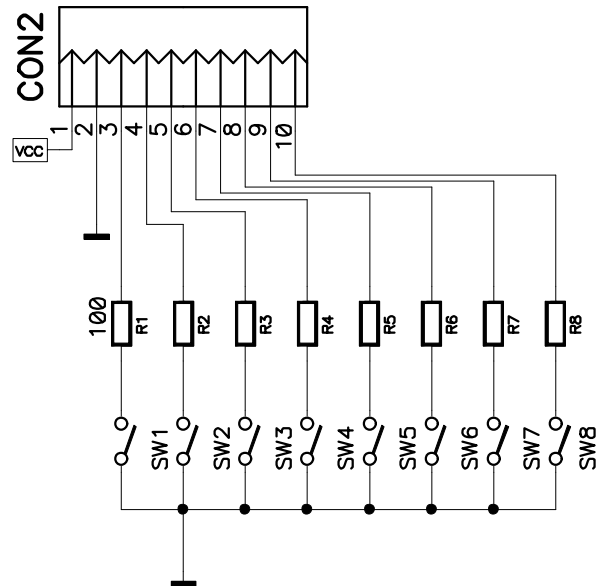


Figure 7

To use the keyswitches, connect CON2 to any I/O port (PortA, PortB, PortC, PortD) with a flat cable.

External Crystal oscillator

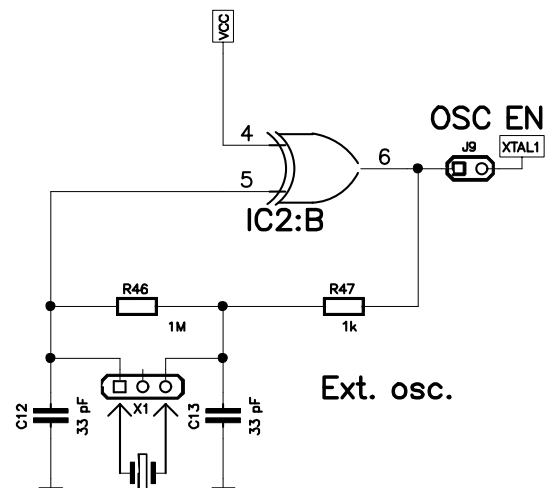


Figure 8

MiniPin II has a built-in External oscillator, where you can mount any crystal from below 1 MHz to up to 16 MHz. The External oscillator can be enabled with jumper J9. When using it with 8-pin AVR's you must also place a jumper at J5.

The external oscillator is very handy to have "on-board" when a microcontroller has its fuse bits

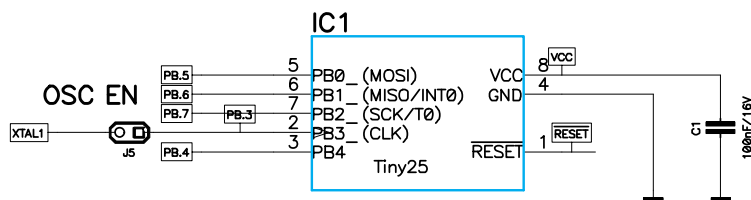


Figure 9

set (usually) accidentally to External Oscillator. With the external oscillator enabled (and a crystal placed in socket X2) we can restore a wrongly-programmed AVR.

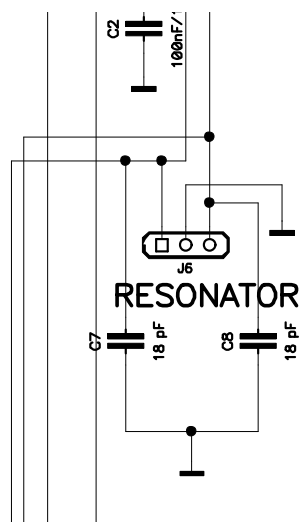


Figure 10

Use of a resonator or Quartz crystal

The MiniPin II board can also use a resonator or XTAL, if it is placed in socket J6. The capacitors needed for use with the crystal are pre-placed on the PCB.

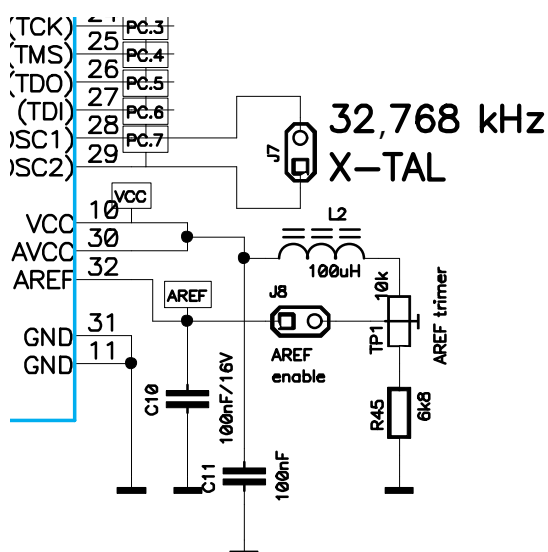


Figure 11

AREF and 32,768 kHz XTAL

The MiniPin II board has a built-in external reference voltage, which can be enabled with jumper J8 (AREF Enable). With the TP1 trimpot, the user can set AREF from nearly 5V down to 2.5 Volts. An external AREF is handy when performing analog to digital conversion (A/D) on voltages between 2.5 and 5.0 Volts. Most AVRs have both a built in Vcc (generally 5.0V) and an accurate 2.5V reference voltage. Using the Vcc reference voltage (generally 5.0V), the resolution of the 10-bit A/D conversion is 4.88 millivolts. If we perform A/D conversions of input voltages in the 3.0 V full-scale range, and we are looking for the best resolution, we would quickly see that a 5V reference is not suitable. With a 3.0 V reference voltage, we can achieve a resolution of 2.93 millivolts, which is nearly twice as good as that with 5.0 V reference voltage.

NOTE

With ATmega8 and ATmega168 connect a 32,768 kHz crystal to socket J6 and set fuse bits to "Ext. Low freq. crystal"!

A 32,768 kHz (watch) crystal is very handy with AVRs that have a Timer2 which can operate in asynchronous mode. With a little Bascom code and 32,768 kHz watch crystal (see RTC chapter), we can make a Real Time Clock with actual date & time and not using too many microcontroller resources. RTC (Real Time Clock) clock routines can also wake-up the AVR microcontroller while in the Powersave mode, thus conserving battery power.

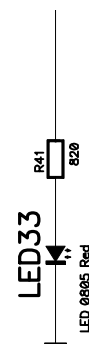


Figure 12

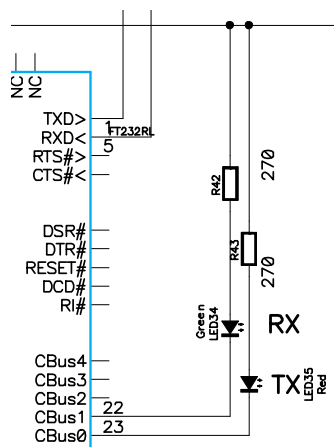


Figure 13

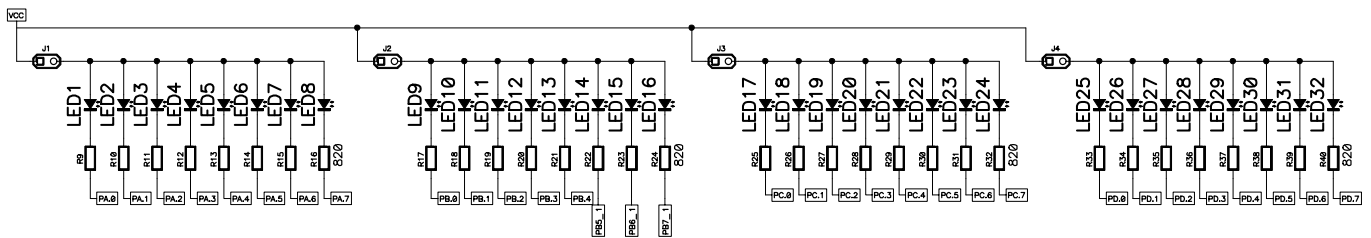


Figure 14

LED signalization

The MiniPin II board features many LEDs:

- » Power supply OK,
- » Rx/Tx communication,
- » 8 LEDs on all four I/O Ports.

Communications

The MiniPin II board supports the following communications methods:

Serial bi-directional communications:

bi-directional communication between the target microcontroller and the USB port using an FT232RL USB bridge.

1-Wire

Place a DS18S20 in any of the sockets J10 to J15, connect CON5 with a flat cable to any Port connector (CON3, CON4, CON6, CON7). Socket J10 corresponds to PortX.0, J11 corresponds to PortX.1 etc. See the 1-Wire sensor section earlier in the chapter for more details.

Example program ID	
Name:	1Wire_1.BAS
Microcontroller:	Any AVR microcontroller
Testing circuit:	Figure 16
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Single 1-Wire sensor on a bus, temperature is displayed on LCD	

Example program ID	
Name:	1Wire_4.BAS
Microcontroller:	Any AVR microcontroller
Testing circuit:	Figure 6
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Multiple 1-Wire sensors, one on each Port	

Example program ID	
Name:	For_loop.BAS
Microcontroller:	Any AVR microcontroller
Testing circuit:	/
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Simple FOR-Next loop for testing LEDs on all ports	

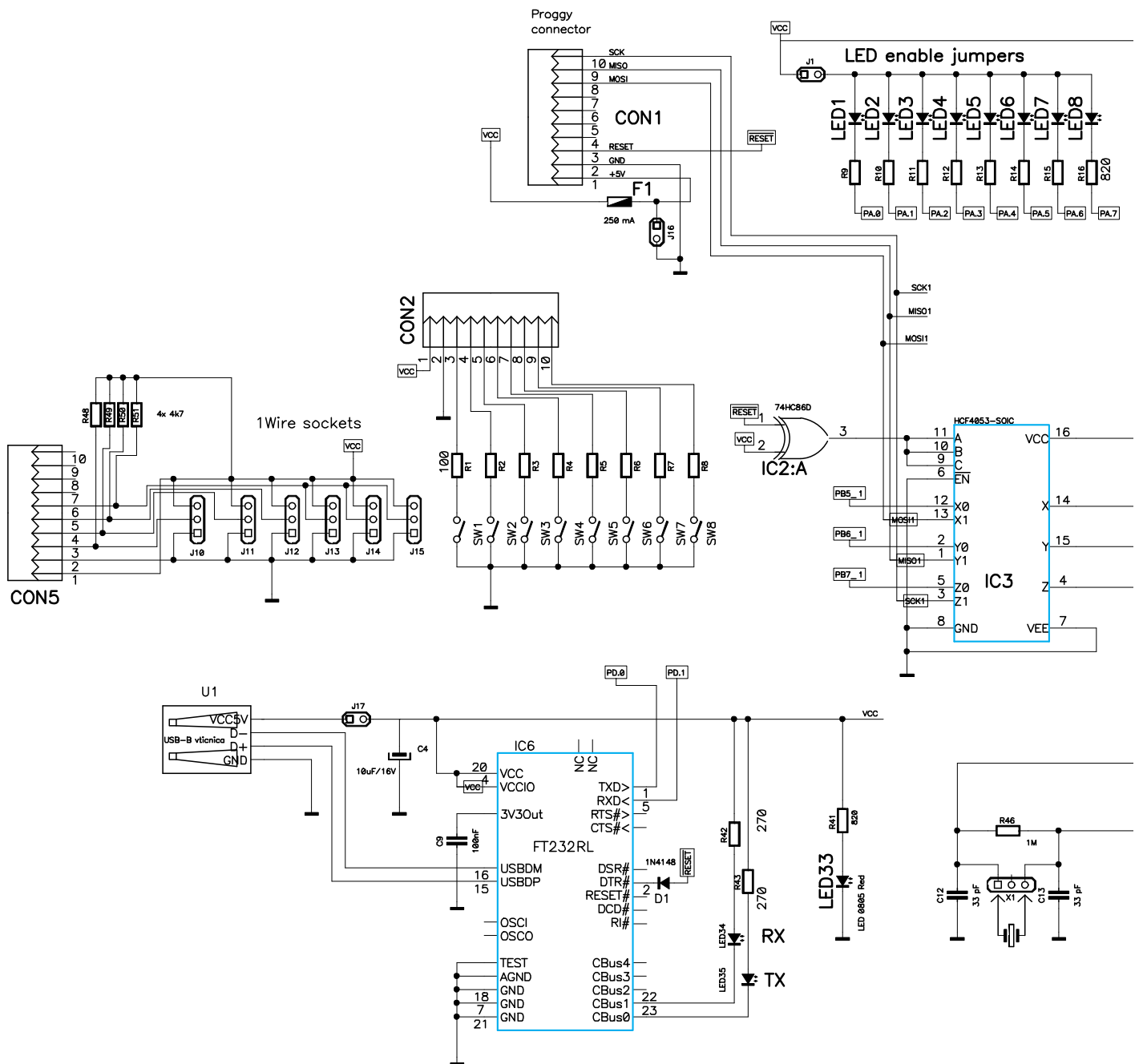
Example program ID	
Name:	GLCD_1.BAS
Microcontroller:	ATmega AVR
Testing circuit:	Figure 1, page 67 (GLCD chapter)
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Use of graphical LCD with table drawn on LCD and 8x8 font	

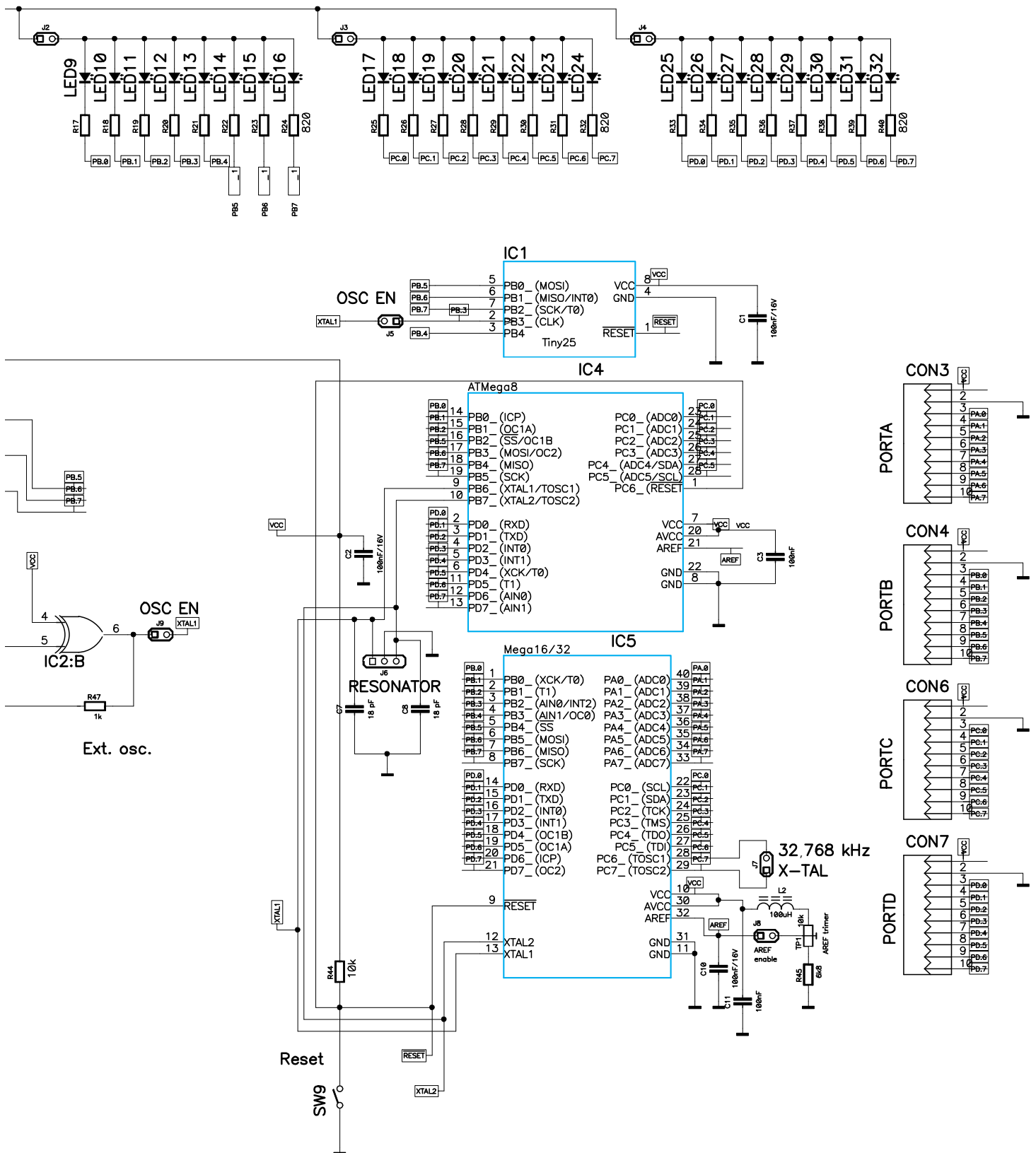
Example program ID	
Name:	1Wire_2.BAS
Microcontroller:	Any AVR microcontroller
Testing circuit:	Figure 6
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Program displays 1-Wire ID of two sensors on the bus	

Example program ID	
Name:	GLCD_2.BAS
Microcontroller:	ATmega AVR
Testing circuit:	Figure 1, page 67 (GLCD chapter)
MiniPin compatibility:	yes
MegaPin compatibility:	yes
GLCD with fonts that can be placed anywhere on GLCD	

Example program ID	
Name:	1Wire_3.BAS
Microcontroller:	Any AVR microcontroller
Testing circuit:	Figure 6
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Same as 1Wire_2.BAS but ID of sensors are stored in EEPROM	

Example program ID	
Name:	GLCD_3.BAS
Microcontroller:	ATmega AVR
Testing circuit:	Figure 1, page 67 (GLCD chapter)
MiniPin compatibility:	yes
MegaPin compatibility:	yes
GLCD with the use of a touch panel including simple calibration of the touch panel	





MiniPin II schematics

Example program ID	
Name:	GLCD_4.BAS
Microcontroller:	ATmega AVR's
Testing circuit:	Figure 1, page 67 (GLCD chapter)
MiniPin compatibility:	yes
MegaPin compatibility:	yes
GLCD with improved calibration for the touch panel	

Example program ID	
Name:	Megaclock1.bas
Microcontroller:	ATmega AVR's
Testing circuit:	Figure 1, page 67 (GLCD chapter)
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Same as Megaclock.bas with simple routines to set hour/min/sec	

Example program ID	
Name:	LCD8x2.bas
Microcontroller:	Any AVR microcontroller
Testing circuit:	Figure 1, page 67 (GLCD chapter)
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Simple program to display characters on blue 8x2 LCD, including PWM for controlling backlight	

Example program ID	
Name:	PWM1.bas
Microcontroller:	ATmega AVR's
Testing circuit:	/
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Simple PWM routine using the hardware PWM in the AVR MCU	

Example program ID	
Name:	Megaclock.bas
Microcontroller:	ATmega AVR's
Testing circuit:	Figure 1, page 67 (GLCD chapter)
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Clock with 32,768 kHz XTAL and 8x2 LCD	

Example program ID	
Name:	USB-ADC2.bas
Microcontroller:	AVR's with ADC
Testing circuit:	/
MiniPin compatibility:	yes
MegaPin compatibility:	yes
Acquires an analogue voltage and converts it with the A/D, then transmits it to both the USB and to LCD. You can see the value of the voltage in the Terminal window within Bascom-AVR	