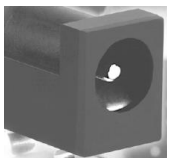
ICP Family Programmers

User's Manual

1 Connectors

1.1 "Power" Connector (Power Jack, Center Pin 2.1mm)



Pin No.	*Pin Name (ICP2-GANG, ICP2-COMBO)*	*Pin Name (ICP2, ICP2(HC), ICP2-Portable)*	*Voltage Range*
1/center	POWER (+)	POWER (+/-)	12V to 15V (9V to 15V for ICP2-Portable)
2	POWER (-)	POWER (+/-)	

1.2 "USB" Connector: Standard Type-B Female

Note: galvanically isolated on ICP2-COMBO "LAN" Connector: Standard RJ-45
Note: available on ICP2-COMBO only

1.3 "RS-232 IN" Connector (Standard D-type 9 Female)

Notes:
- not available on ICP2-Portable
- galvanically isolated on ICP2-COMBO

Pin No.	*Pin Name*	*Voltage Range*	*Pin Type*	*Description*
1	-	-	-	Not connected
2	PC_RXD	RS-232 level	RS-232 output	TxD output to PC
3	PC_TXD	RS-232 level	RS-232 input	RxD input from PC
4	PC_DTR	-15V to +15V	Power	**ICP2-GANG**: Not connected **ICP2/ICP2(HC)**: Additional power supply input
5	GND	-	GND	Ground connection
6	12V_OUT	11-14VDC	Power	**ICP2-GANG**: power output **ICP2/ICP2(HC)/ICP2-COMBO**: Not connected
7,8,9	-	-	-	Not connected

1.4 "RS-232 OUT" Connector (Standard D-type 9 Male)

Note: available on ICP2-GANG only

Pin No.	*Pin Name*	*Voltage Range*	*Pin Type*	*Description*
1	-	-	-	Not connected
2	CHAIN_232_RXD	RS-232 level	RS-232 input	RxD input from next ICP2-GANG
3	CHAIN_232_TXD	RS-232 level	RS-232 output	TxD output to next ICP2-GANG
4	-	-	-	Not connected
5	GND	-	GND	Ground connection
6,7,8,9	-	-	-	Not connected

1.5 "Control Interface" Connector (DIN-64 A,B Male)

Available on ICP2-COMBO only (Contact Softlog Systems for details):
- galvanically isolated
- standalone operation without PC (GO/PASS_OUT/FAIL_OUT, environment select)
- chain connection (RS-232 in/out)

1.6 "TARGET" Connector D-type 15 Female: all programmers excluding ICP2-COMBO

Notes: ICP2-GANG - 4 identical channels
ICP2/ICP2(HC)/ICP2-Portable - 1 channel

Pin No.	*Pin Name*	*Voltage Range*	*Pin Type*	*Description*
1	T_VDD	2.0V to 5.5V	Output or input with weak pull-down and programmable strong pull-down	Target VDD supply voltage
2	GND	-	-	Ground connection
3	T_SCK	2.0V to 5.5V	CMOS output or input with weak pull-down	Target clock
4	T_MOSI	2.0V to 5.5V	CMOS output or input with weak pull-down	Target data
5	T_MISO	2.0V to 5.5V	CMOS output or input with weak pull-down	Target data, internally connected to T_MOSI
6	T_VPP/MCLR	2.0V to 13.5V	Output or input with weak pull-down	Target VPP supply voltage
7	T_TARG **(2)**	5.0V	CMOS output	General purpose output
8	T_VTEST	2.0V to 13.5V	Output	Target VTEST signal for PIC17Cxxx family
9	T_DIO_0	2.0V to 5.5V	CMOS output or input with weak pull-down	Target VPP output for LVP or FOSC signal for PIC17Cxxx family
10	T_DIO_1	2.0V to 5.5V	CMOS output or input with weak pull-down	Target PGM output for LVP
11	GND	-	-	Optional ground connection
12	GND **(1,4)**	-	-	Optional ground connection
13	GO **(1,3,4)**	0-1.0V or N/C	CMOS input with pull-up 10K	Input for programming activation in standalone mode
14	PASS_OUT**(1,3,4)**	5.0V	CMOS output	Output for pass/fail/busy indication
15	FAIL_OUT **(1,3,4)**	5.0V	CMOS output	Output for pass/fail/busy indication

Notes:
(1) Dedicated for standalone operation without PC
(2) Not available on ICP2-Portable
(3) Optional for ICP2-Portable, contact Softlog Systems for details
(4) ICP2-GANG: presents at each TARGET connector (4 total)

1.7 ICP2-COMBO only: "TARGET" Connector (DIN-48, A,B,C, male), 3 Identical Connectors

Pin Number				*Pin Name*	*Opto-relay barrier*	*Pin Type*	*Description*
CH. 1 CH. 5 CH. 9	*CH. 2 CH. 6 CH. 10*	*CH. 3 CH. 7 CH. 11*	*CH. 4 CH. 8 CH. 12*				
A1	A5	A9	A13	T_VPP/ MCLR	Yes	Power output or input with weak pull-down	Target VPP/MCLR supply voltage
A2	A6	A10	A14	T_SCK	Yes	CMOS output or input with weak pull-down	Target clock
A3	A7	A11	A15	T_MOSI	Yes	CMOS output or input with weak pull-down	Target data
A4	A8	A12	A16	T_DIO_0	Yes	CMOS output or input with weak pull-down	Target VPP output for LVP or FOSC signal for PIC17Cxxx family
B1	B5	B9	B13	T_DIO_1	Yes	CMOS output or input with weak pull-down	Target PGM output for LVP
B2	B6	B10	B14	T_MISO	Yes	CMOS output or input with weak pull-down	Target data, internally connected to T_MOSI
B3	B7	B11	B15	T_VTEST	Yes	Output	Target VTEST signal for PIC17Cxxx family
B4	B8	B12	B16	T_VDD	Yes	Power output or input with weak pull-down and programmable strong pull-down	Target VDD supply voltage
C1	C5	C9	C13	GND_SW	Yes	-	Ground connection via opto-relay barrier
C2, C3	C6, C7	C10, C11	C14, C15	GND	-	-	Ground connection (permanent)
C4	C8	C12	C16	T_TARG	-	CMOS output	Optional output to control target power supply

1.8 Typical Connection to "TARGET" Connector

Programmer Pin Name	*PIC10/12/16/18/24 dsPIC®/PIC32*	*PIC17*	*I2C*	*Keeloq®*
T_VDD	VDD	VDD	VDD	VDD
GND	GND	GND	GND	GND
T_SCK	CLOCK (PGC, PGEC)	CLOCK (PGC)	SCL	CLOCK
T_MOSI	DATA (PGD, PGED)	DATA (PGD)	SDA	DATA
T_MISO	-	-	-	-
T_VPP/MCLR	MCLR/VPP	MCLR/VPP	-	-
T_TARG	-	-	-	-
T_VTEST	-	VTEST	-	-
T_DIO_0	-	FOSC	-	-
T_DIO_1	-	-	-	S1

IMPORTANT: In order to use Enhanced ICSP™ for dsPIC33/PIC24 devices, a pull-down resistor 3.3K-10K Ohm must be placed between T_MOSI (PGD) and GND. If your PCB contains a PGD pull-up resistor then value of the resistor should be about 20% of the pull-up resistor but not less than 1.5K Ohm. For more info contact Softlog Systems: support@softlog.com

2 PC-Driven and Standalone Modes

ICP family programmers can be operated in PC-driven and/or standalone mode

Programmer	***PC-Driven***	***Standalone***
ICP2/ICP2(HC)	Yes	Yes
ICP2-GANG	Yes (single channel only)	Yes
ICP2-COMBO	Yes (single channel only)	Yes
ICP2-Portable	Yes	Yes
ICP-01	Yes	No

PC-driven mode means that all programming parameters and data are set in **PC** and the PC executes required sequences (programming, verification, blank check, etc)

Standalone mode means that all programming parameters and HEX file data ("Environment") are saved in **programmer's** non-volatile flash memory. See paragraph 14 "Preparing Environment and Transferring Environment to Programmer".

Standalone programming can be activated by 2 ways:
- from PC
- by GO input on the programmer unit (NOTE: optional on ICP2-Portable)

Simultaneous multi-channel programming can be done in standalone mode only

3 Chain Connection (ICP2-GANG and ICP2-COMBO)

Number of channels can be increased (up to 64 channels) by daisy chain connection between ICP2-GANG or ICP2-COMBO units.

ICP2-GANG: see ***"ICP2-GANG Quick Start"*** manual for details
ICP2-COMBO: contact Softlog Systems for details

IMPORTANT: all programmer units should have the **same** (all=yes or all=no) "DLL/Command Line Support" option

4 PASS/FAIL LEDs and Outputs

###	*Conditions*	*PASS LED*	*FAIL LED*	*PASS Output*	*FAIL Output*
1.	Power-up	2 sec ON		2 sec ON	
2.	Operation in-progress (busy)	ON		ON	
3.	Programming done: **PASS**	ON	OFF	ON	OFF
4.	Programming done: **FAIL** (verification error)	OFF	ON	OFF	ON
5.	UUT problem during operation: - Vdd overload - Vpp overload - I2C communication error	OFF	Blink	OFF	ON
6.	Non-UUT problem during standalone operation: - database error - device not supported - no Keeloq® support - no dsPIC® support - etc.	OFF	Blink	OFF	OFF
7.	No firmware presents (bootloader only)	Slow blink	OFF	OFF (not supported)	
8.	Firmware upgrade in-progress	Fast blink	OFF	OFF (not supported)	

5 Other LEDs (ICP2-Portable)

See ***"ICP2-Portable Quick Start.pdf"***

6 Standalone Operation without PC

- Prepare an environment and transfer to programmer – see paragraph 14 for details
  NOTE: once the environment is saved in **non-volatile** memory it's automatically ready for programming
- Short pin GO to GND for at least 100ms to start programming
- Observe PASS/FAIL LEDs or/and pins PASS_OUT and FAIL_OUT – see paragraph 4 for details

7 Host Computer Requirements

- Pentium-4 or greater IBM PC compatible
- Resolution 1024x768 or higher
- 256MBytes of RAM
- Windows-XP/Vista/7/8. Contact Softlog Systems for operation with Win-95/98/NT/2000
- At least 200MBytes of hard disk space
- CD-ROM drive
- Free RS-232 port (all programmers excluding ICP2-Portable) or USB port

8 Installation

8.1 Important Note

In the past the default directory was specified as "C:\Program Files\Soft-Log\...
Starting from version 4.13.1a Jan-2015 ICP family software will be installed to a default directory C:\Softlog\ which allows to avoid virtual storage of CFG and INI files

8.2 Preliminary Installation

8.2.1 Software Installation

To install the software supplied, follow the steps below:

- Insert ICP family CD in the CD-ROM drive. An opening screen appears
- Click "Install ICP for Windows" and follow the on-screen instructions

If the opening screen does not appear:
- Double-click on the "My Computer" icon
- Double-click the icon for your CD-ROM drive
- Double-click "Icp_CD.exe"

8.2.2 Preliminary Hardware Installation

- Connect the programmer to its power supply (not required for ICP2-Portable)
- Connect RS-232 or USB cable between PC and the programmer
- Install USB driver according to ***"ICP2 USB Driver Installation"*** manual
  NOTE: USB driver installation is not required for operation with RS-232 port

8.3 ICP2-GANG Setup

- Install ICP2-GANG according to ***"ICP2-GANG Quick Start"*** manual

8.4 ICP2-COMBO Setup

- Install ICP2-COMBO according to ***"ICP2-COMBO Quick Start"*** manual

8.5 ICP2-Portable Setup

- Install ICP2-Portable according to ***"ICP2-Portable Quick Start"*** manual

8.6 ICP2/ICP2(HC) Software Setup

8.6.1 Run "ICP_Win.exe" Program

- Double-click "ICP_Win" icon
- Press "Yes" if message "Newer firmware is available. Upgrade now?" appears

8.6.2 Run "Programmer/Quick Start Wizard" and follow the Wizard

9 Plug-in to MPLAB® IDE

- ICP2-GANG and ICP2/ICP2(HC) programmers are integrated into Microchip MPLAB® IDE.

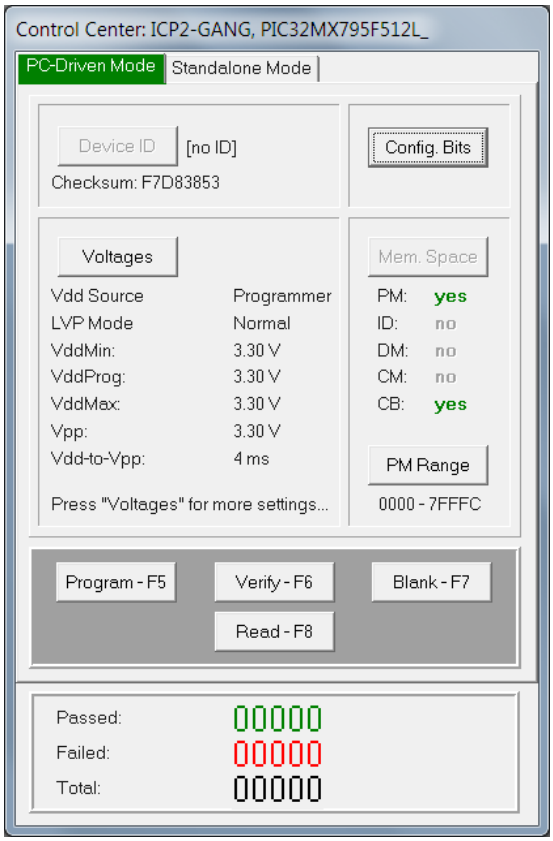
10 Checksum Calculation and Programming Buffers

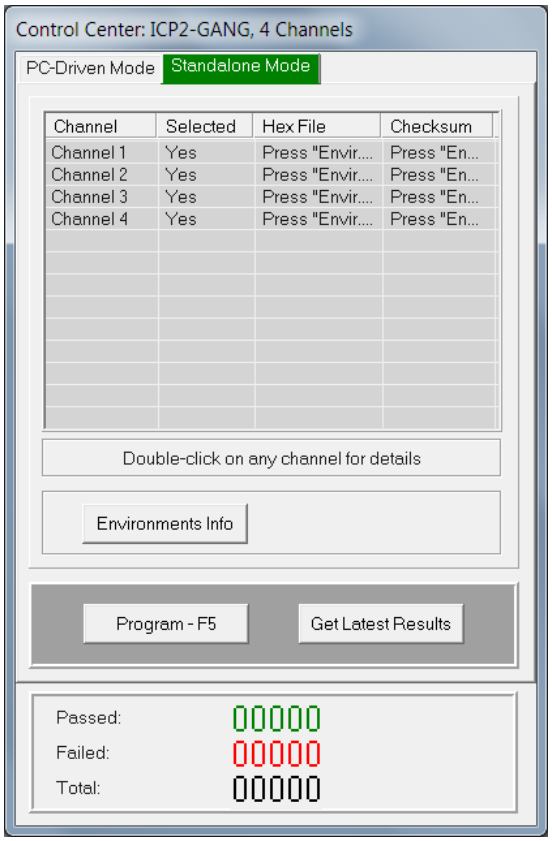
ICP2 family software calculates HEX file checksum (CS) as follows:
- **Unprotected** CS is calculated according to Microchip® programming specifications (as MPLAB or MPLAB X)
- In contrary to the programming specifications, **protected** CS is calculated as unprotected one. MPLAB/MPLAB X calculates its according to the specifications that makes the result CS nearly unusable (flash CS is not calculated at all)

Note that ICP2 software intentionally **doesn't** clear programming buffers (it allows to merge partial HEX files), therefore CS may change if you load one HEX file after another

11 Control Center

Control Center has 2 operation modes: PC-driven and Standalone





11.1 Control Center in PC-Driven Mode

Control Center in PC-driven mode allows the following operations:

- Edit device ID
- Edit configuration bits of the device
- Select memory space
- Set PM range
- Programming, Verification, Blank Check and Read

11.2 Control Center in Standalone Mode

Control Center in standalone mode allows the following operations:

- Get environment information for all channels (button Environment Info)
- View environment details of selected channel (double-click on selected channel)
- Programming
- Get latest results

12 Menu Commands

12.1 File Menu (Alt-F)

Open (Import)...	Open a HEX file from disk and load it into buffer memory area
Save	Save the currently loaded file
Save As (Export)...	Save the buffer to a HEX file on disk
Save Configuration	Save all current settings
Exit	Exit the software

12.2 Edit Menu (Alt-E)

Edit/Fill Program Memory	Fill an area of the Program Memory with a specified value
Edit/Fill Data Memory	Fill an area of the Data Memory (EEPROM) with a specified value
Read-only Editors	Enable/disable edit of Program and Data Memory buffers

12.3 Environment (Alt-P)

Save Environment As…	Save current setup and buffers in environment format (*.pj2)
Transfer Environment to Programmer…	Transfer Environment (*.pj2) to programmer
Environment Wizard…	

12.4 Serialization Menu (Alt-S)

Disable	Disable serialization
Load File	Load serialization file
Create File	Create serialization file

12.5 Device Menu (Alt-D)

Select a type of device to be programmed

12.6 Programmer Menu (Alt-G)

Select Programmer	Select programmer (ICP2/ICP2(HC) or ICP2-GANG)
GANG Configuration	Select active GANG channels (64 max)
Assign Address to GANG Box	Assign address to currently connected programmer (1-16). See ***"ICP2-GANG Quick Start"*** for details
Quick Start Wizard	

12.7 Run Menu (Alt-R)

Program	PC-driven mode: program data in the buffer(s) into the device Standalone mode: activate standalone programming
Verify	Verify the data in the device against the data in the buffer(s)
Blank Check	Check the data in the device for the blank state
Read	Read the device and store the data in the buffers
Program Only	Open a window for repeated programming

12.8 Communication Menu (Alt-C)

RS-232/USB/LAN COM	Select the desired COM port
Connect	Connect to the programmer

12.9 Options Menu (Alt-O)

Voltage	Set desired voltages
Clock/Data/MCLR(Advanced)	Set desired Clock/Data/MCLR parameters
Preferences	Select options for programming
Firmware Upgrade	Execute firmware upgrade
Activation of Options	Execute activation of optional components: - DLL/Command Line Support (D) - dsPIC®/PIC24 Support (P) - Keeloq® Support (K) - Secure Programming Support (S) - PIC32 Support (X) Note: **Contact Softlog Systems for activation details**

12.10 Speed Optimization Menu (Alt-T)

Run Speed Optimization Utility	Run a utility that automatically configures the programmer settings for the optimal speed performance
Optimization Summary	Show speed-related summary

12.11 Help Menu (Alt-H)

Read me	Display "Readme_w.txt" file
About	Connect with the programmer and display software and firmware versions

13 Shortcuts

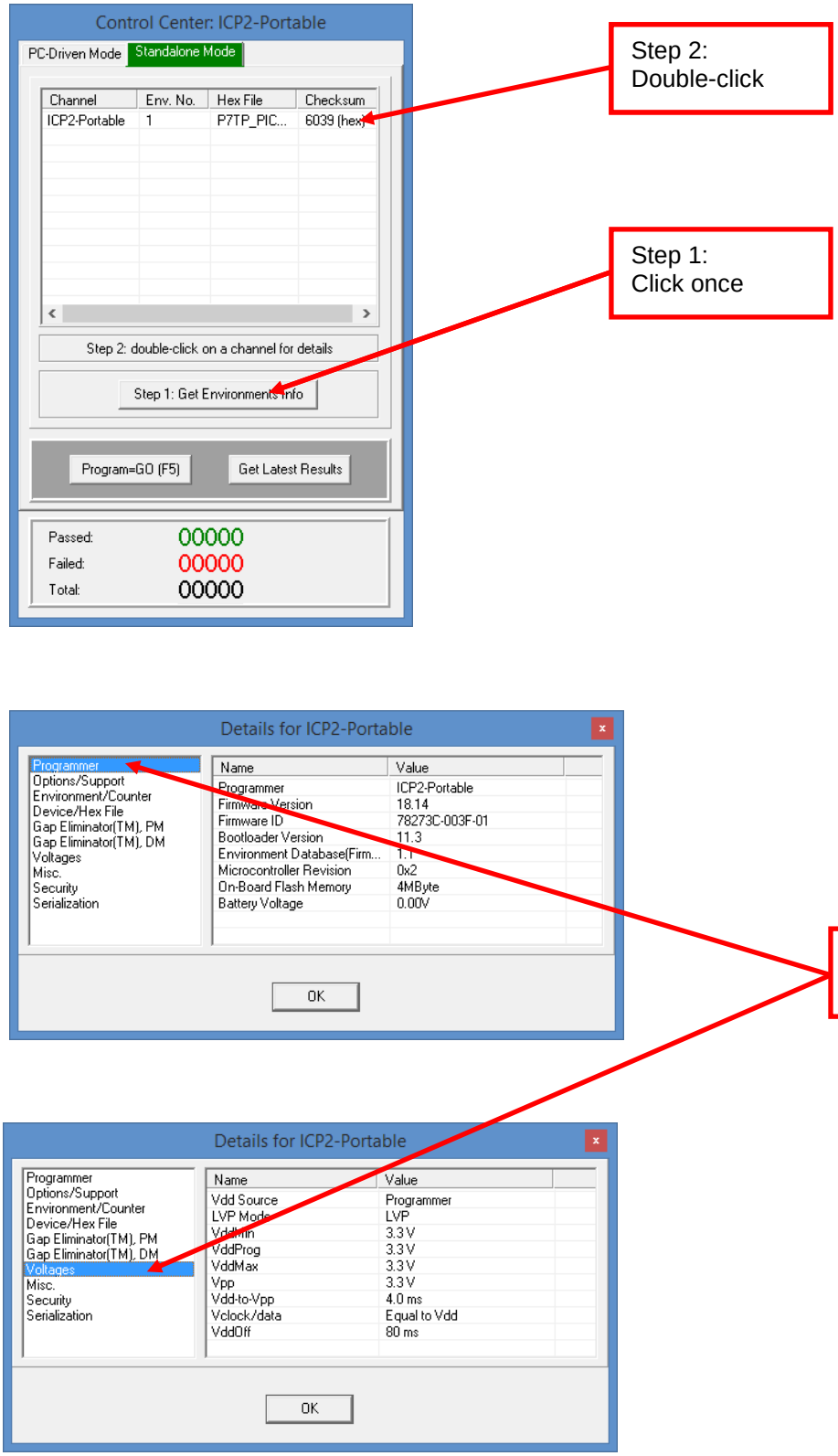
Save	Ctrl-S
Open...	Ctrl-O
Program	F5
Verify	F6
Blank Check	F7
Read	F8
Programming Only	F9
Menu	F10
Close Window	Ctrl-F4
Exit	Alt-F4

14 Preparing Environment and Transferring Environment to Programmer

- Run "Environment/Environment Wizard" and follow the Wizard
- Select programmer and press "Next"
- ICP2-GANG/ICP2-COMBO only: select GANG channels and press "Next"
  ICP2-Portable only: select environment number and press "Next"
- Select Device
  From the "Device" list select a device to be programmed and press "Next"
- Set Voltages and press "Next"
- Load (open) a HEX file.
  NOTE: The programmer software is able to read ID information, data memory (EEPROM) contents and configuration bits from the HEX file
- Save Environment
  - Press on "…" button
  - Type in environment name, 16 characters max
  - Press "Save"
  - Press "Next"
- Transfer Environment to Programmer
  - Press on "Transfer Environment" button, select your environment and press "Open"
  - Wait until environment is transferred to all channels
  - Press "Next"
- Switch to Standalone Mode
  - Press on "Standalone Mode" button
  - Press "Finish"
- Your system is ready for standalone programming
- View the transferred environment as shown in Paragraph 15

15 Viewing Environment

An environment inside the programmer can be viewed as shown below



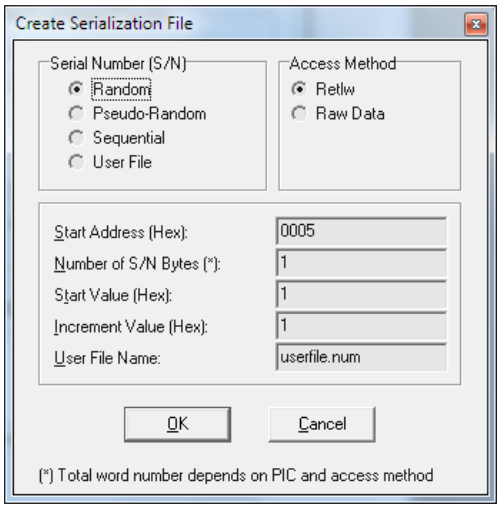
16 Serialization

16.1 Create Serialization File

- Select "Serialization/Create File" to generate a serialization file

- Enter the following data:
  - Serial Number (serialization scheme): random, pseudo-random, sequential and user file
  - Start Address. The address should be valid for the device
  - Number of S/N Bytes. Enter number of bytes (1 to 8) for your serial number

    WARNING: Total number of result **words** (addresses) depends on the device and access method
  - Start Value. Enter the start value (1 to 16 hex digits). If the start value is greater than the maximum value for the number of bytes selected the most significant digits will be truncated. The start value must differ from zero for pseudo-random scheme.
  - Increment Value. Valid for the sequential scheme only
  - User File Name. Valid for "user file" scheme only
  - Access Method. Select Retlw or Raw Data



- Press OK to save a serialization file

NOTES:
- A currently selected serialization file will be updated after any successful programming for single-channel programming and after any programming attempt for ICP2-GANG
- The "retlw" opcode ("retlw" access method) will be automatically generated for a selected type of devices, i.e.:
  - 08(Hex) for low-end microcontrollers (12C5xx, etc)
  - 34(Hex) for mid-range microcontrollers (16C/Fxxx)
  - b6(Hex) for high-end microcontrollers (17C7xx)
  - 0c(Hex) for enhanced microcontrollers (18Fxxx)
  - 054(Hex) for 16-bit devices (pattern: 0000_0101_0100_kkkk_kkkk_dddd)

16.2 Serialization File Example 1

SerializationScheme = 2 (0-Random, 1-Pseudo-Random, 2-Sequential, 3-User File)
StartAddress = 0005 (Hex)
NumberOfWords = 2 (Hex)
CurrentValue = 0000000000001234 (Hex)
IncrementValue = 1 (Hex)
UserFile = userfile.num
AccessMethod = 0 (0-retlw,1-raw data)

The following program memory locations will be updated as follows:

- PIC16xxx
  0005: 3434
  0006: 3412

- PIC12C5xx
  0005: 0834
  0006: 0812

16.3 Serialization File Example 2 (User File Scheme)

SerializationScheme = 3 (0-Random, 1-Pseudo-Random, 2-Sequential, 3-User File)
StartAddress = 0005 (Hex)
NumberOfWords = 2 (Hex)
CurrentValue = 0000000000001234 (Hex)
IncrementValue = 1 (Hex)
UserFile = File1.num
AccessMethod = 0 (0-retlw, 1-raw data)

User file should contain serial numbers in HEX radix, for example:
1111
2222
3333
4444
5FC1

The user file will be updated by placing semicolon (;) at very beginning of the string, for example:
; 1111
; 2222
3333
4444
5FC1

If your numbers start from very beginning of the string the 1-st digit will be replaced by semicolon:
;111
;222
3333
4444
5FC1

16.4 Enable Serialization

Select "Serialization/Load File" to activate serialization

16.5 Disable Serialization

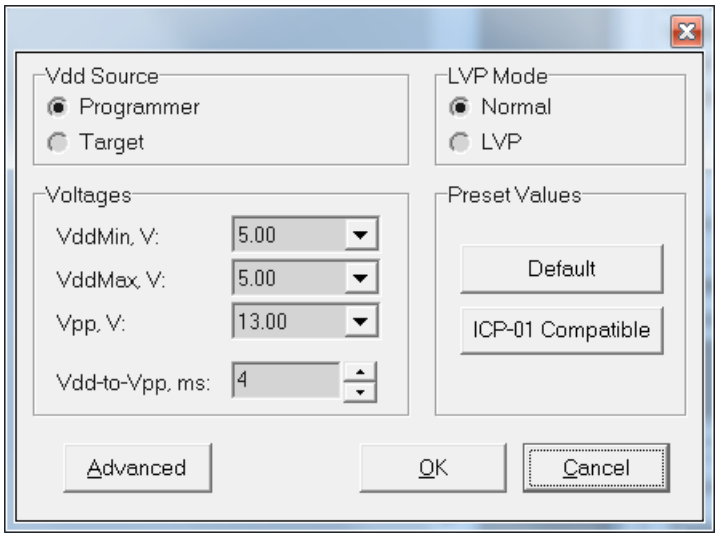
The serialization will be disabled in the following cases:
- "File/Open..." command is executed
- "Serialization/Disable" command is executed
- "Edit/Read-only Editors" is set to edit mode
- "Run/Read" command is executed
- a new device is selected
- user's serialization file is empty
- Control Center switches between PC-driven and standalone modes

16.6 Standalone Serialization

- Make all settings (select device, voltages, etc.)
- Load a HEX file
- Select "Serialization/Load File" to activate serialization
- Create an environment by "Environment/Save Environment As…"
- Transfer the environment to programmer

17 Voltages

17.1 Menu: Options → Voltage



17.2 Vdd Source and LVP Mode

The programmer executes operations at the following Vdd voltages

###	***Vdd Source***	***LVP Mode***	***Vdd during Programming***	***Vdd during Verify***	***Vdd during Blank Check***	***Vdd during Read***
1.	Programmer	Normal	Database	VddMin, VddMax ***(Note 2)***	VddMin	Database
2.	Programmer	LVP	VddMax ***(Note 1)***	VddMax	VddMax	VddMax
3.	Target	Normal	Target	Target	Target	Target

Notes:
1) Use LVP mode if you want to change default programming voltage
2) Set VddMin=VddMax to disable the 2-nd verification pass

17.3 Vpp Voltage

The Vpp voltage is the same for all the operations

17.4 Vdd-to-Vpp Delay

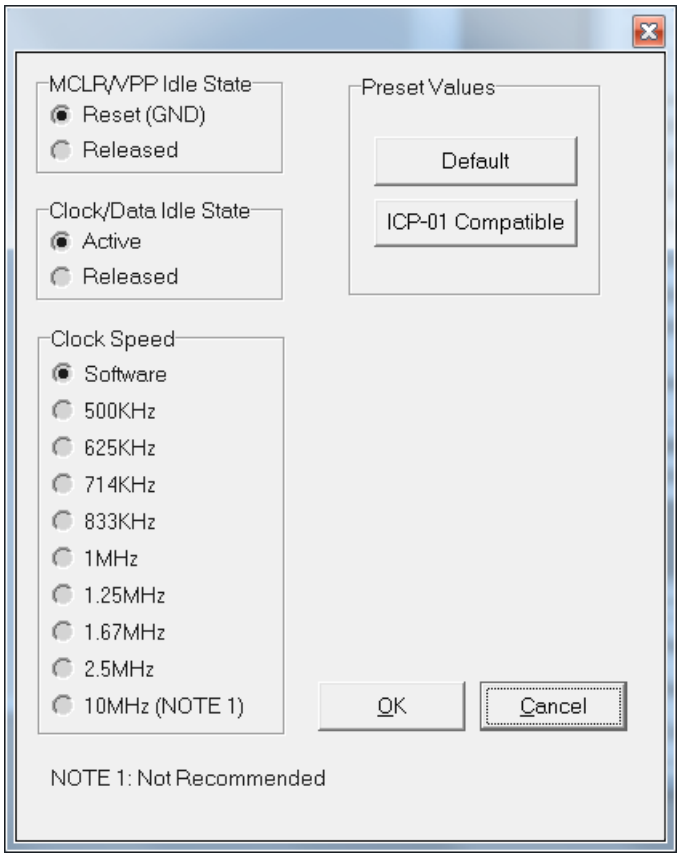
Delay between Vdd and Vpp can be in range 0.1...250ms. It is recommended to use default delay of 4ms to correctly enter the programming mode. Longer delays may be useful if the Vdd line has high capacitance (more than 200uF) which causes the Vdd to rise slowly

17.5 ICP-01 Compatibility

Press on "ICP-01 compatible" button forces the following settings:
- Vdd Source: Programmer
- LVP Mode: Normal
- Voltages: Database values

18 Clock, Data and MCLR/VPP

18.1 Menu: Options → Clock/Data/MCLR(Advanced)



18.2 MCLR/VPP Idle State

Reset (GND): Programmer permanently keeps MCLR in reset state (GND) when no operation
Released: Programmer releases MCLR with weak pull-down of about 160KΩ

18.3 Clock/Data Idle State

Active: Programmer configures data/clock pins as outputs when no operation
Released: Programmer releases data/clock with weak pull-downs of about 300KΩ

18.4 Clock Speed

Clock speed can be selected for enhanced microcontrollers (PIC18F) and 16-bit devices (PIC24, dsPIC30 and dsPIC33). It's recommended to use high clock speed (2.5MHz) for devices with memory size bigger than 32K and dsPICs. **NOTE:** 10MHz is not recommended

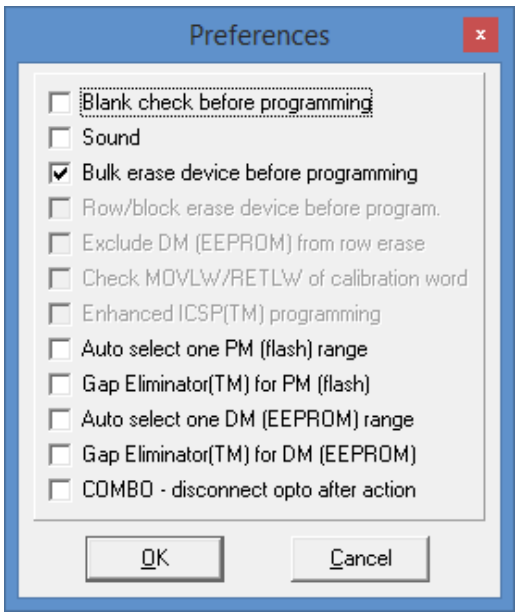
18.5 ICP-01 Compatibility

Press on "ICP-01 compatible" button forces the following settings:
- MCLR/VPP Idle State: Reset (GND)
- Clock/Data Idle State: Active
- Clock Speed: Software (100-500KHz, depends on device family)

19 Preferences

19.1 Menu: Options → Preferences

NOTE: some items are grayed out if they are not supported by a selected device



19.2 Blank check before programming [ ]

Enables/disables blank check operation before device programming. This option is not useful for flash devices

19.3 Bulk erase device before programming [x]

When the option is ON the device will be automatically erased by bulk erase mechanism

IMPORTANT:
- this option must be set to ON for proper operation with most of devices
- it's the only option to erase code protected device

19.4 Row erase device before programming [ ]

When the option is ON the device will be automatically erased by the row erase mechanism

IMPORTANT:
- row erase can't erase a code protected device
- available for dsPIC30 family and several mid-range devices only (PIC16F1xxx, PIC16F81x, etc.), may be useful for operation at low voltages

19.5 Exclude DM (EEPROM) from row erase [ ]

When the option is ON the DM (EEPROM) is excluded from row erase procedure
NOTE: available for dsPIC30 family only, may be useful to preserve EEPROM

19.6 Check MOVLW/RETLW of calibration word [x]

When the option is ON an opcode of the calibration memory is tested during programming
NOTE: available for devices which have a calibration word with movlw/retlw opcode (PIC12F519, PIC12F675, etc.)

19.7 Enhanced ICSP™ Programming [x]

When the option is ON the device is programmed/verified using Enhanced ICSP™ method (much faster)

NOTE: available for most of dsPIC33/PIC24 devices. PIC32 always uses Enhanced ICSP™ therefore this option is grayed when PIC32 is selected

IMPORTANT: a pull-down resistor is required for Enhanced ICSP™ of dsPIC33/PIC24 devices – see paragraph 1.8

IMPORTANT - Enhanced ICSP™ limitations (Microchip® silicon issues):
- PGEC3/PGED3 programming pair does not work on several devices – check Microchip® errata
- Enhanced ICSP™ may not work if "Windowed WDT" is enabled

19.8 Auto select one PM (flash) range [ ]

When the option is ON one optimum PM range is selected. This feature is similar to "Allow programmer to select memories and ranges" from other programmer/debugger manufacturers

19.9 Gap Eliminator™ for PM (flash) [x]

When this option is ON ICP2 family programmer automatically excludes **multiple** empty (blank) PM areas in the HEX file from the programming process, resulting in shortened programming time. See paragraph 20 for more details

19.10 Auto select one DM (EEPROM) range [ ]

When the option is ON one optimum PM range is selected. This feature is similar to "Allow programmer to select memories and ranges"

IMPORTANT: Due to different silicon read/write protection mechanisms the DM (EEPROM) may be not erased before programming for several devices, therefore test your device before going to production. If it is then don't enable this feature

19.11 Gap Eliminator™ for DM (EEPROM) [x]

When this option is ON ICP2 family programmer automatically excludes **multiple** empty (blank) DM areas in the HEX file from the programming process, resulting in shortened programming time. See paragraph 20 for more details

IMPORTANT: Due to different silicon read/write protection mechanisms the DM (EEPROM) may be not erased before programming for several devices, therefore test your device before going to production. If it is then don't enable this feature

19.12 COMBO: disconnect opto after action [ ]

When the option is ON the ICP2-COMBO opto-relay barrier will be disconnected after programming/verification/blank check/read

20 Gap Eliminator™

20.1 Overview

The Gap Eliminator™ enables end customers to exclude empty (blank) flash and EEPROM areas in the HEX file from the programming process, resulting in shortened production cycles and reduced manufacturing costs. This powerful feature is available in all of Softlog's in-circuit programmers

20.2 How It Works

In addition to the critical data they carry, HEX files may also contain multiple empty areas (gaps). These gaps may come at the beginning, in the middle, or at the end of the HEX file. Thus, when programming a microcontroller, the empty bytes of a HEX file are also burned onto the microcontroller. In order to "skip" these gaps, a typical programmer usually allows the operator to define a single programming range, thus enabling two empty areas to be skipped at the beginning and end of the file. However, if the gap(s) are located between valid data areas (see example below), this is not an effective solution.

Softlog's Gap Eliminator™ feature solves this problem. Before a production run, it automatically analyzes the HEX file and effectively removes multiple gaps (up to five) from the Program Memory (flash) and Data

Memory (EEPROM). This significantly reduces programming time for mass production operations.

20.3 Example of HEX File with Gaps

Memory size = 4096 bytes (address range 0x0000…0x0FFF)

Gaps are highlighted in yellow; valid data in green

```
Address 0x0000…0x0007: FF FF FF FF FF FF FF FF
…
Address 0x0078…0x007F: FF FF FF FF FF FF FF FF
Address 0x0080…0x0087: 01 53 A4 67 88 A5 CD 6F
Address 0x0088…0x008F: 01 23 45 67 89 AB CD EF
…
Address 0x01F8…0x01FF: 51 F3 45 F7 89 A6 CC CF
Address 0x0200…0x0207: FF FF FF FF FF FF FF FF
Address 0x0208…0x020F: FF FF FF FF FF FF FF FF
…
Address 0x07F8…0x07FF: FF FF FF FF FF FF FF FF
Address 0x0800…0x0807: 01 53 A4 67 88 A5 CD 6F
Address 0x0808…0x080F: 01 23 45 67 89 AB CD EF
…
Address 0x09F8…0x09FF: 51 F3 45 F7 89 A6 CC CF
Address 0x0A00…0x0A07: FF FF FF FF FF FF FF FF
Address 0x0A08…0x0A0F: FF FF FF FF FF FF FF FF
…
Address 0x0FF0…0x0FF7: FF FF FF FF FF FF FF FF
Address 0x0FF8…0x0FFF: 01 53 A4 67 88 A5 CD 6F
```

As noted, a typical programmer allows you to define one range 0x0080…0x0FFF (3968 bytes), skipping the empty area at the beginning of the file. **This reduces the size of the HEX file by 128 bytes (3.2%).**

Using a Softlog ICP programmer with the Gap Eliminator™ feature, three programming ranges can be defined for this example:

- 0x0080…0x01FF (384 bytes)
- 0x0800…0x09FF (512 bytes)
- 0x0FF8…0x0FFF (8 bytes)

This effectively eliminates all the gaps and **reduces the size of the HEX file to be programmed by 3,192 bytes (77.9%)**.

21 Speed Optimization Utility

The Speed Optimization Utility is a wizard that guides the user through the ICP2 programmer configuration settings to ensure optimal speed performance. These settings include Clock/Data speed, Vdd-to-Vpp delay, VddOff delay, Gap Elimination, Enhanced ICSP™ and more.

Enter "Speed Optimization → Run Speed Optimization Utiliy"

22 Configuration File

The ICP setup is saved in a configuration file named "icp01.cfg".

IMPORTANT: the program reads a configuration file that is located in a directory which specified in "Start in" property. This approach allows creation of unlimited configurations on the same PC

Normally, a configuration file should **not** be modified by a text editor.

23 Command Line Parameters (GUI)

Some parameters can be loaded from the command line:

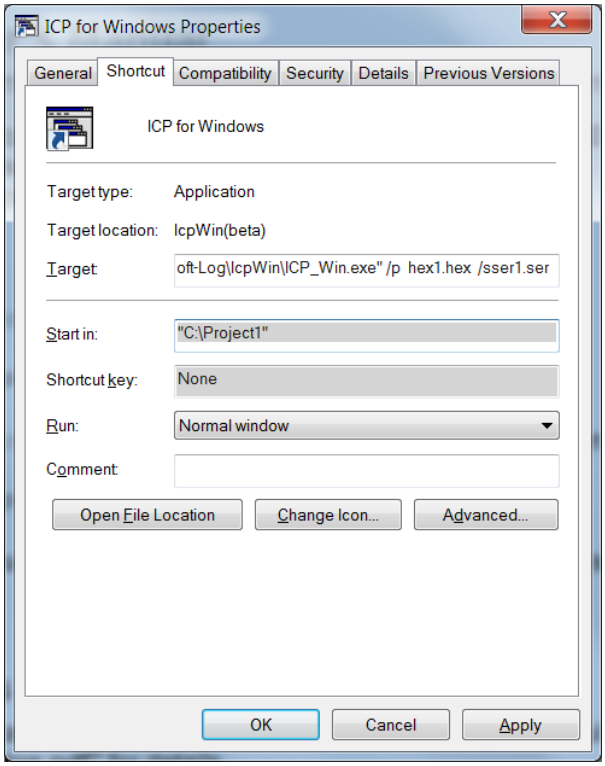
<Hex file> - hex file to be loaded
/c<Configuration file> - configuration file to be loaded, overwrites local "icp01.cfg"
/s<Serialization file> - serialization file to be loaded
/p - production mode (one-touch operation)

Examples:

- Start in the production mode and load file "hex1.hex":
  <path to ICP_Win.exe> /p hex1.hex

- Start in the production mode and load hex file "hex1.hex" and serialization file "ser1.ser":
  <path to ICP_Win.exe> /p hex1.hex /sser1.ser

  "Start in" property should specify a directory where "hex1.hex" and "ser1.ser" are located



24 DLL Functions

ICP family programmers can be run from the user's application using powerful set of DLL functions. See document ***"DLL Description.pdf"*** for details

25 Command Line Interface (non-GUI)

ICP family programmers can be run from the user's application using full-featured command line interface. See document ***"ICP Command Line.pdf"*** for details

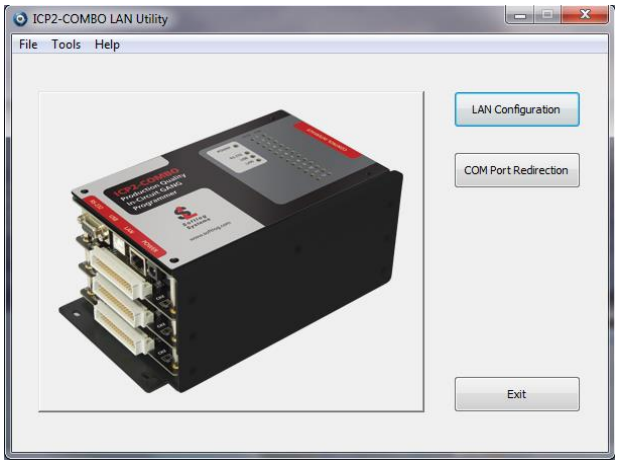
26 Secure Programming

Your hex files contain business-critical intellectual property that could be compromised during the contract manufacturing process. Utilizing patent pending technology, our Secure Programming feature provides several layers of protection that dramatically reduce the risk of unauthorized reconstruction of hex data. See document ***"Secure Programming Utility User's Manual.PDF"*** for more details

27 Standard LAN Configuration

To install and run the software supplied, follow the steps below:

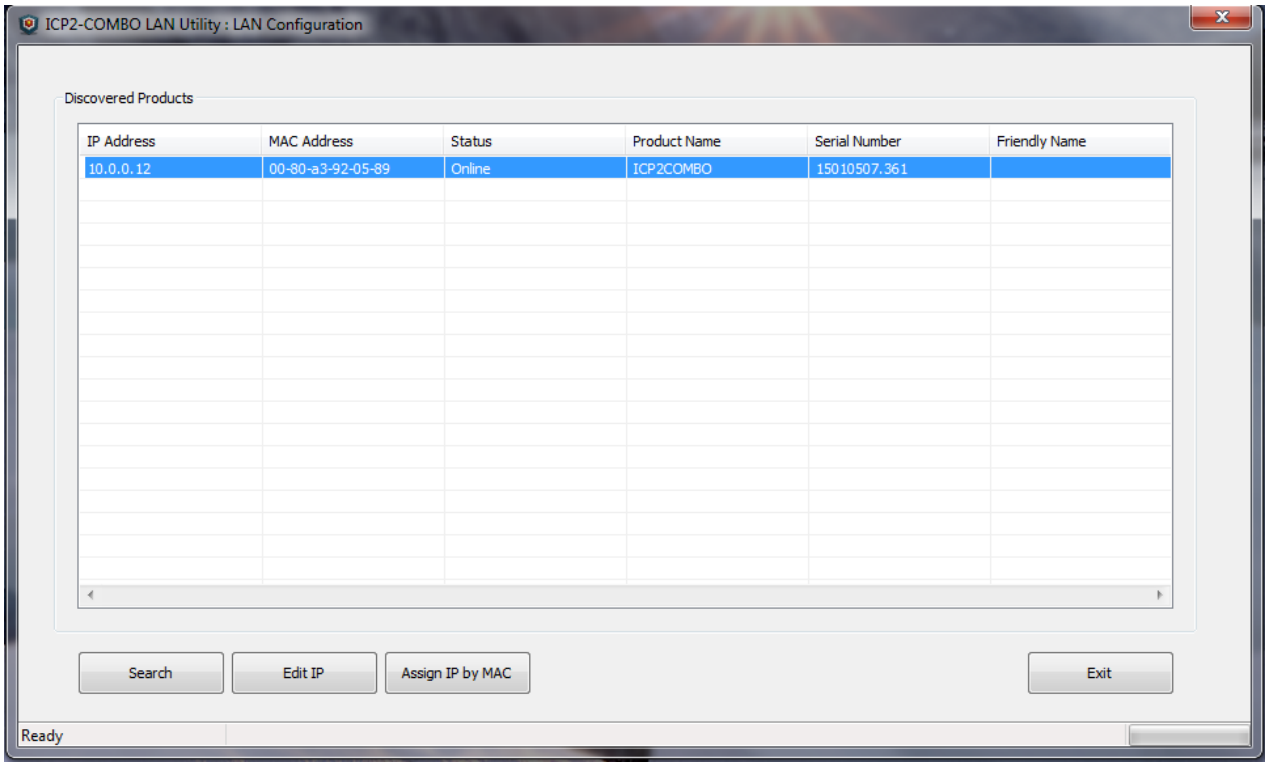
- Insert ICP family CD in the CD-ROM drive. An opening screen appears
- Click "ICP LAN Configurator" and follow the on-screen instructions
- Connect ICP2-COMBO programmer to the same Ethernet network as your PC:
  - use straight cable for connection via network hub or switch
  - use crossover cable for direct connection to the PC
- Run "ICP-LAN" application that is located under C:\Softlog\IcpLan. An opening screen appears:



The "ICP-LAN" application provides 2 configuration features:
- "LAN Configuration" which discovers ICP2-COMBO programmers and changes IP network parameters
- "COM Port Redirection" which creates a virtual COM (CPR) port for ICP2-COMBO programmer

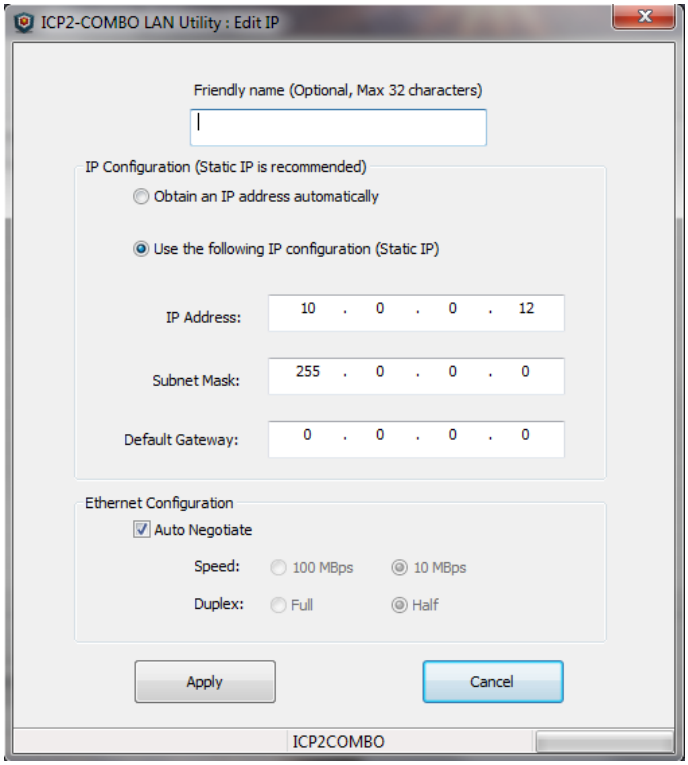
27.1 LAN Configuration

- Click "LAN Configuration" button
- The application automatically starts discovering programmers. You can retry by clicking "Search" button
- The following screen appears:



NOTES:
- verify that detected product serial number is the same as printed on the ICP2-COMBO sticker
- if your device was not found refer to "Advanced IP Configuration" below (paragraph 28.1)
- if status of discovered device is "Unreachable", reconnect the power to the device and retry searching, otherwise refer to "Advanced IP Configuration".

- Select (highlight) the discovered ICP2-COMBO programmer
- Click "Edit IP" button. The following screen appears:



- Change your settings if required

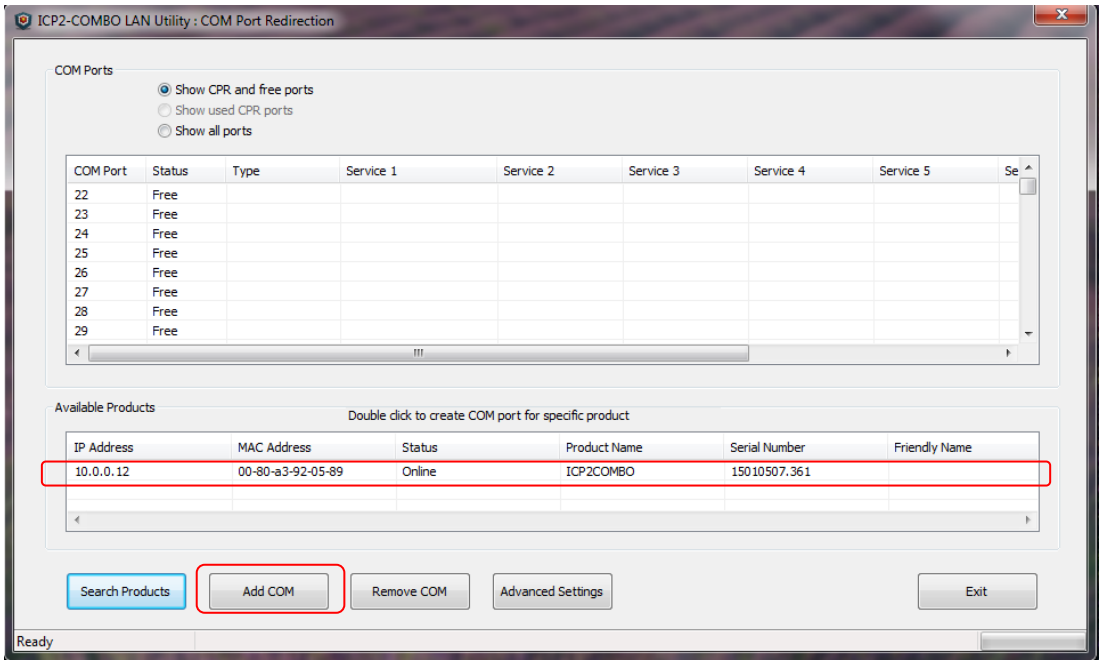
  NOTES - the following settings are recommended:
  - assign a friendly name to ICP2-COMBO, for example "COMBO-12 for tester 4"
  - use static IP for convenient operation with a virtual CPR COM port

- Press "Apply". Wait until operation is complete (may take about 1 minute)

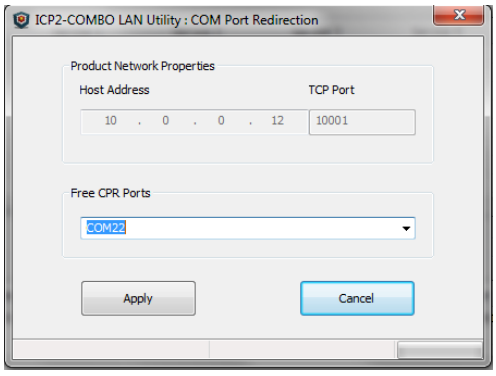
- Press "Exit"

27.2 COM Port Redirection

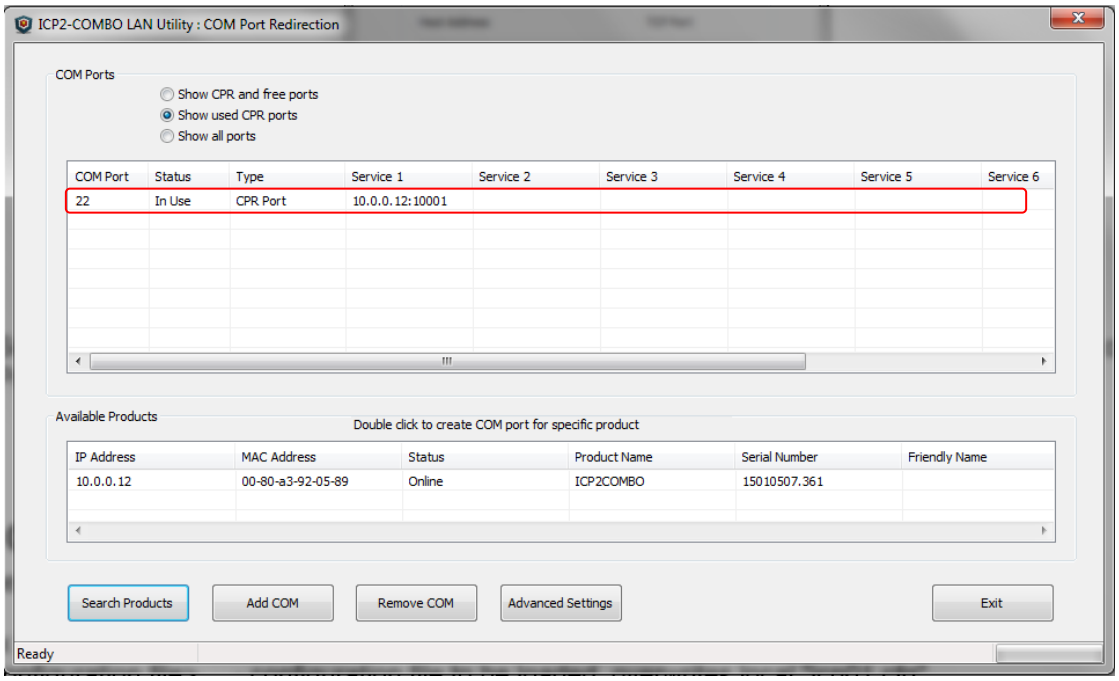
- Click "COM Port Redirection" button (from the opening screen). The following screen appears:



- Select (highlight) the discovered ICP2-COMBO programmer on the bottom list
- Click "Add COM" button. The following screen appears:



- Select desired COM port number and press "Apply". COM Ports list on "COM Port Redirection" window will refresh automatically:



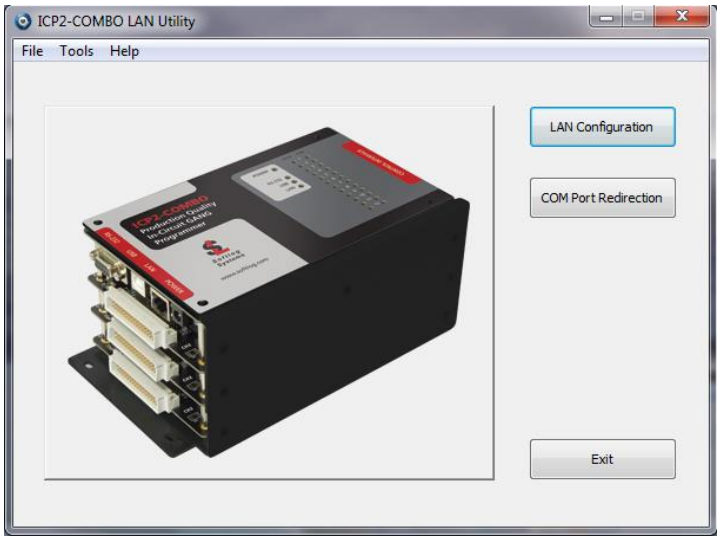
- Press "Exit"

28 Advanced LAN Configuration

28.1 Advanced IP Configuration

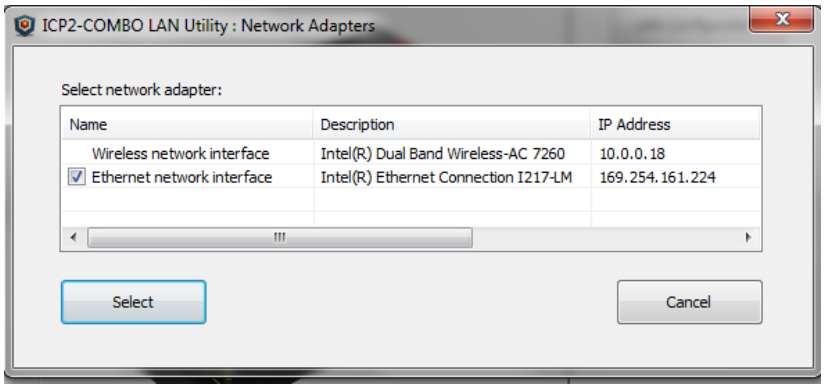
If your ICP2-COMBO programmer is not found or is unreachable, perform the following steps:

- Run "ICP-LAN" application



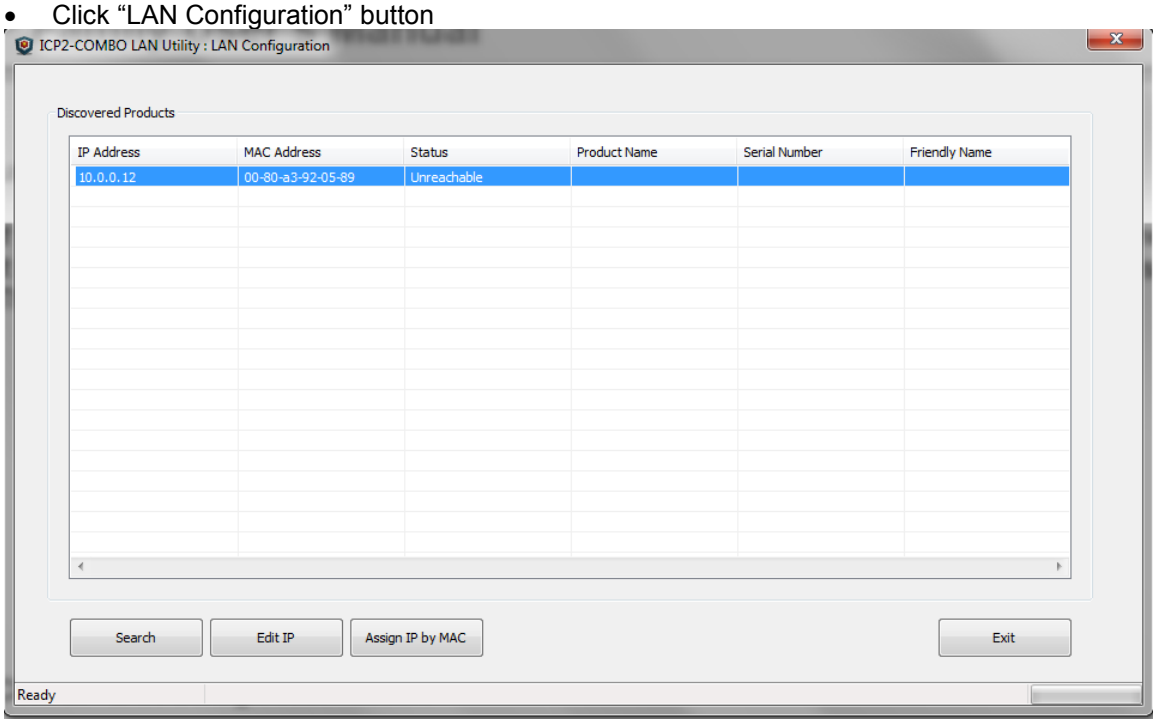
- If your PC has several network adapters, select the adapter which ICP2-COMBO programmer is connected to:
  select Tools → Options → Select Network

- The following screen appears:

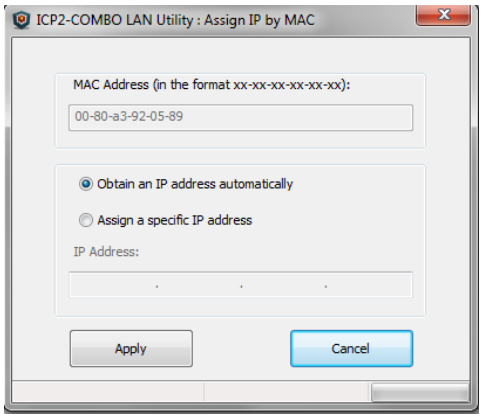


- Select desired network adapter and press Select

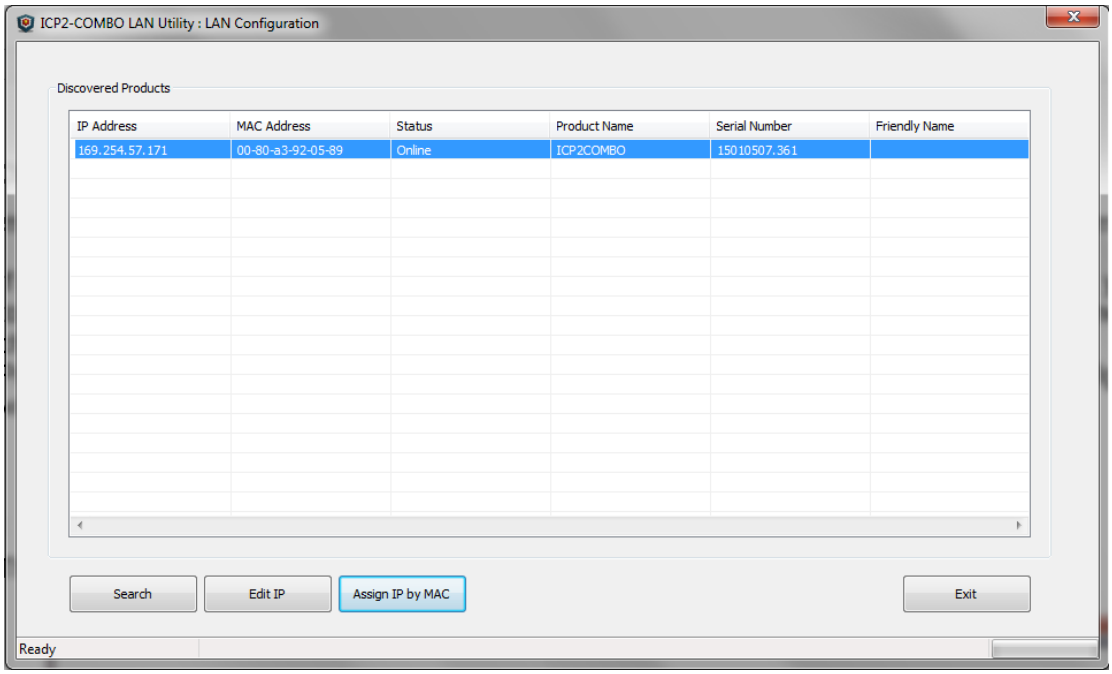
- Click "LAN Configuration" button



- Click "Assign IP by MAC" button:

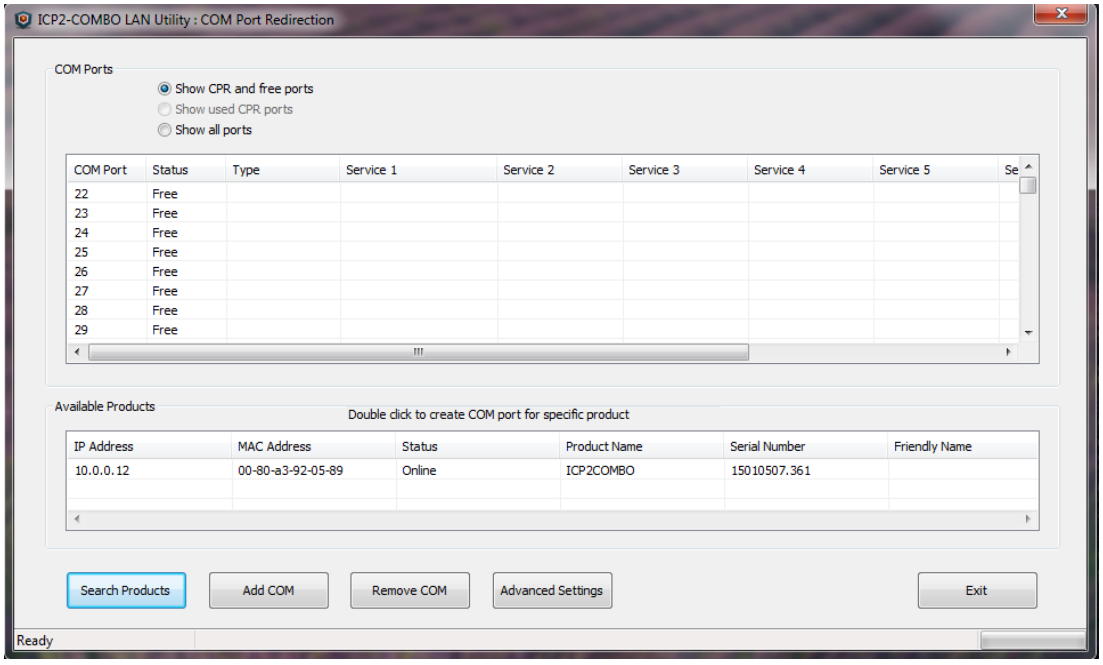


- Enter MAC address manually according to sticker printed on the ICP2-COMBO chassis
- It is recommended to choose "Obtain an IP address automatically" radio button
- Press "Apply" button. Wait until the procedure is finished (it may take about one minute)
- Once done, "LAN Configuration" window refreshes its list

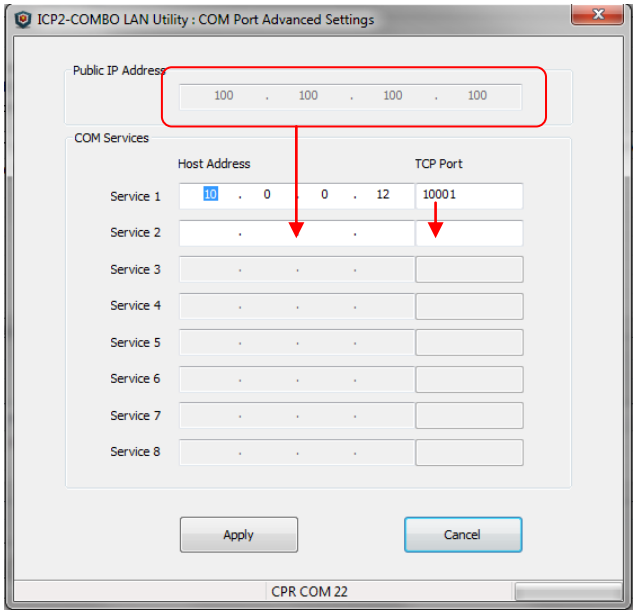


28.2 Advanced CPR Configuration

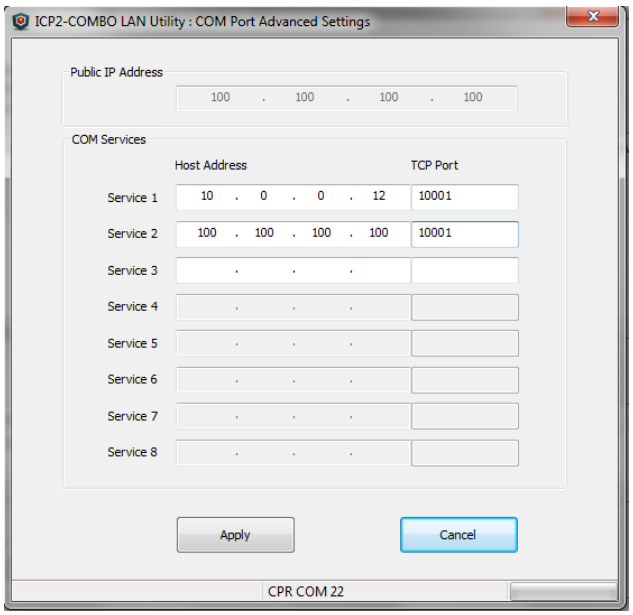
There is a possibility to apply additional services to the created CPR port or to change parameters of the existing service. For example, you can add service with host address equal to public (external) IP address of your Ethernet network in order to access ICP2-COMBO programmer remotely. Up to 8 services can be applied to a CPR port.



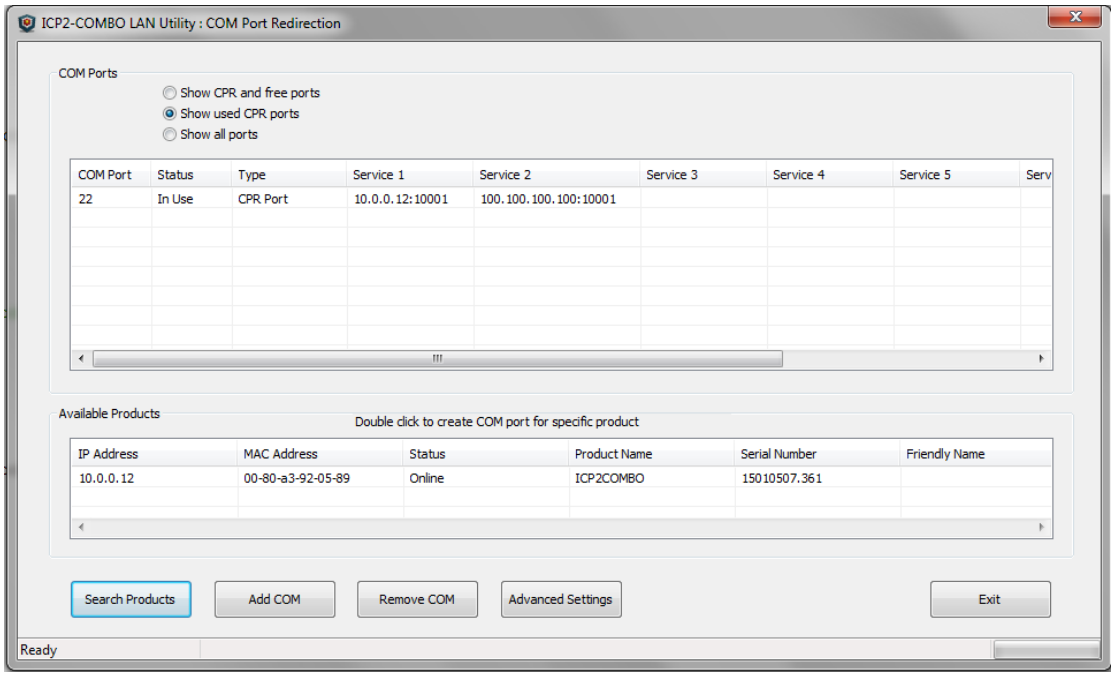
- Select (highlight) the discovered product on the bottom list
- Click "Advanced Settings". The following screen appears:



- Copy public IP address and TCP port as shown above



- Press "Apply"



- Press "Exit"

29 Manual Production Mode (One-Touch Operation)

The production mode is a powerful option for volume programming

The following steps should be done to correctly prepare the software for programming in the production mode:

- Create a subdirectory (C:\FILE_HEX)
- Copy your CFG, HEX and SERIALIZATION files to FILE_HEX subdirectory (for example: "hex1.hex" and "ser1.ser")
  NOTE: serialization file is optional
- Change ICP_Win shortcut property "Start in" to C:\FILE_HEX
- Change ICP_Win shortcut property "Target" to C:\...\ICP_Win.exe hex1.hex /sser1.ser /p
- Double-click ICP_Win icon for programming

The program will be terminated in the following cases:
- Communication error
- Hex file error
- Serialization file should be loaded (/s appears) but loading is failed

30 In-Circuit Programming

Standard in-circuit programming is done through 5 wires (VDD, GND, CLOCK, DATA and VPP)

30.1 Vdd

- Maximum Vdd current consumption by the application circuit :
  - ICP2/ICP2-GANG/ICP2-COMBO: 250mA
  - ICP2(HC): 1000mA
  - ICP2-Portable: 100mA

- Maximum Vdd capacitance: 1000-10000uF. For ICP2/ICP2-GANG/ICP2-COMBO, increase Vdd-to-Vpp delay by about 20ms for every 1000uF

NOTE: If your circuit has low current consumption (less than 10mA) in conjunction with high capacitance (more than 100uF), the load resistor (100-510 Ohm) must be connected between Vdd and GND pins of the programmer for faster discharge of Vdd capacitor

30.2 Vpp

Vpp recommended load (ICP2/ICP2-GANG/ICP2-COMBO/ICP2-Portable): > 1KOhm, < 33nF
Vpp recommended load (ICP2(HC)): > 100 Ohm, <100nF

WARNING: due to high VPP requirements ICP2-Portable may not be suitable for OTP devices

30.3 CLOCK and DATA

CLOCK/DATA recommended load(ICP2/ICP2-GANG/ICP2-COMBO/ICP2-Portable): > 3.3KOhm, < 33pF
CLOCK/DATA recommended load(ICP2(HC)): > 50 Ohm, < TBD nF

30.4 Delay between Vdd and Vpp

This delay should be as short as possible (default 4ms is recommended)

NOTE: for more details contact Softlog Systems to obtain ***"ICP2-GANG Specification"***, ***"ICP2-COMBO Specification"***, ***"ICP2 Specification", "ICP2(HC) Specification" and "ICP2-Portable Specification"***

31 Target Cable

Softlog Systems recommends to use the following cable between ICP2 family programmer and target device:

- Length: as short as possible, less than 50cm
- Structure: unshielded separate wires
- Low-pass filter: 22-47pF between CLOCK and GND **as close as possible** to the target microcontroller is recommended, especially for long cables (> 20cm)

32 Appendix A: Power Supply

The ICP system (excluding ICP2-Portable) is shipped with its own power supply. If the user wishes to connect his/her own power, make sure the following specifications are met:

Programmer	*Output Voltage*	*Output Current*	*Center Terminal, 2.1mm*
ICP2-GANG	12VDC	1.5A	"+"
ICP2-COMBO	12VDC	5A	"+"
ICP2	12VDC	0.5A	"-" or "+"
ICP2(HC)	12VDC	1.5A	"-" or "+"
ICP2-Portable	9-15VDC	0.5A	"-" or "+"

33 Appendix B: Return Values (Errorcodes)

```
enum AUTO_ERROR_LEVEL { //return values
AUTO_OK                 = 0,  //operation OK
AUTO_DB_ERR             = 1,  //database error
AUTO_COM_ERR            = 2,  //communication error
AUTO_VDD_ERR            = 3,  //Vdd overload error
AUTO_VPP_ERR            = 4,  //Vpp overload error
AUTO_HEX_ERR            = 5,  //HEX file loading error
AUTO_SER_ERR            = 6,  //serialization file error
AUTO_VER_ERR            = 7,  //verification error
AUTO_ERR_NO_SPACE       = 8,  //no space selected
AUTO_SAVE_ERR           = 9,  //file save error
AUTO_SOCK_ERR           = 10, //socket communication error (obsolete)
AUTO_I2C_ERR            = 11, //UUT I2C communication error
AUTO_DLL_ERR            = 12, //DLL programming is not supported
AUTO_KEY_ERR            = 13, //key generation error
AUTO_CFG_ERR            = 14, //config. file error
AUTO_COM_NUM_ERR        = 15, //invalid COM number
AUTO_COM_BUSY_ERR       = 16, //selected COM is busy
AUTO_COM_BAUD_ERR       = 17, //invalid baud rate
AUTO_COM_NO_OPEN        = 18, //can't open COM port
```

```
AUTO_USER_CANCEL            = 19, //user cancel
AUTO_IN_PROGRESS            = 20, //operation in progress
AUTO_BC_ERR                 = 21, //blank check error
AUTO_OP_NOT_ALLOW           = 22, //operation not allowed for selected programmer
AUTO_FW_INVALID             = 23, //firmware invalid-firmware upgrade needed
AUTO_24LC_ADDR_ERR          = 24, //24LC01 address (offset) is out of range
AUTO_DM_ADDR_ERR            = 25, //DM range error
AUTO_FIRM_ERR               = 26, //firmware version error
AUTO_NO_SUB                 = 27, //no ICP-SUB PCB
AUTO_NO_SUP_KEE             = 28, //no keeloq support
AUTO_NO_SUP_DSPIC           = 29, //no dsPIC support
AUTO_ICP2_REQ               = 30, //ICP2 required
AUTO_DEV_ERR                = 31, //device selection error (unspecified error)
AUTO_PROG_MISMATCH          = 32, //mismatch between selected and detected programmers
AUTO_PRJ_INVALID            = 33, //Invalid environment
AUTO_PRJ_DB_FIRM_PC_MIS     = 34, //mismatch between PC and firmware database
AUTO_PRJ_DB_FIRM_AT45_MIS   = 35, //mismatch between environment and firmware database
AUTO_DLL_SUPPORT_REQIURED   = 36, //obsolete: "GO" pressed on hardware and no DLL/standalone support
AUTO_PRJ_CS                 = 37, //environment CS error
AUTO_STA_IDLE               = 38, //programmer is idle or standalone operation can't be started
AUTO_STA_BUSY               = 39, //standalone operation: programmer busy
AUTO_ENV_ERR                = 40, //environment file error
AUTO_PM_RANGE               = 41, //invalid PM range specified
AUTO_SEC_SUPPORT_REQUIRED   = 42, //Security support required
AUTO_SEC_CNT_INTEG          = 43, //Future: Security feature: integrity error in counter
AUTO_SEC_CNT_ZERO           = 44, //Future: Security feature: counter = 0
AUTO_SEC_NO_FUNC            = 45, //Future: Security feature: function does not exist
AUTO_SEC_PACK_ERR           = 46, //Future: Security feature: packet error
AUTO_SEC_EEPROM_FAIL        = 47, //Future: Security feature: EEPROM error
AUTO_SEC_ANTI_SCAN          = 48, //Future: Security feature: anti-scan activated,
AUTO_SEC_SEC_ID_CMP         = 49, //Future: Security feature: incorrect Security ID
AUTO_SEC_PASSW_CMP          = 50, //Future: Security feature: incorrect password
AUTO_SEC_BATCH_CMP          = 51, //Future: Security feature: incorrect batch
AUTO_SEC_VERS_ERR           = 52, //Future: Security feature: version error
AUTO_SEC_UNKNOWN_ERR        = 53, //Future: Security feature: unknown error
AUTO_NO_ROW_ERASE           = 54, //row erase is not supported
AUTO_INVALID_PARAM          = 55, //invalid parameters
AUTO_MOVLW_RETLW_CALIB      = 56, //no movlw in calibration word
AUTO_NO_USUAL_ENV_TRAN      = 57, //Usual environment can't be sent if a secure one inside
AUTO_SEC_BUF_START_ADDR     = 58, //sec. buf. properties error: incorrect start addr
AUTO_SEC_BUF_END_ADDR       = 59, //sec. buf. properties error: incorrect end addr
AUTO_SEC_BUF_PAGE_START     = 60, //sec. buf. properties error: incorrect page start
AUTO_SEC_BUF_PAGE_SIZE      = 61, //sec. buf. properties error: incorrect page size
AUTO_SEC_BUF_NOT_EVEN       = 62, //sec. buf. properties error: length not even
AUTO_SEC_BUF_NO_DM          = 63, //sec. buf. properties error: no DM in PIC
AUTO_SEC_BUF_LAST_PAGE      = 64, //sec. buf. properties error: last PM page can't be used
AUTO_SEC_BUF_NO_16BIT_SUP   = 65, //sec. buf. properties error: no Script 1 for 16-bit devices
AUTO_SEC_BUF_NOT_MODULO_3   = 66, //sec. buf. properties error: length not modulo 3
AUTO_SEC_EMPTY_MASK         = 67, //Security feature: empty mask for secure environment
AUTO_TEST_COM_NO_SUPPORT    = 68, //ICP2 test command not supported
AUTO_TEST_NACK              = 69, //ICP2 test command returns NACK
AUTO_NO_SUP_P32             = 70, //no PIC32 support
AUTO_PIC32_BUSY_OR_DAMAGED  = 71, //PIC32 is busy or damaged
AUTO_PIC32_CP_OR_DAMAGED    = 72, //PIC32 is code protected or damaged
AUTO_PIC32_PE_ANSWER        = 73, //PIC32 programming executive: no answer
AUTO_PIC32_PE_VERSION       = 74, //PIC32 programming executive: incorrect version
AUTO_SEC_BUF_NO_32BIT_SUP   = 75, //no security support for PIC32
AUTO_CNT_ZERO               = 76, //non-secure (low-endurance) counter is 0
AUTO_SQTP_CONFLICT          = 77, //serialization from PC is not allowed if standalone serialization=ON
AUTO_INVALID_DEVICE_CFG     = 78, //invalid device number in CFG file. Use latest DLL
AUTO_DEV_ID_NO_SUPPORT      = 79, //Device ID read is not supported for the family
AUTO_ROW_PM_RANGE           = 80, //invalid PM range due to row size
AUTO_PE_MISMATCH            = 81, //Programming executive: mismatch between environment and firmware
AUTO_PE_NO_PGD_PULLDOWN     = 82, //No pull-down on PGD line
AUTO_PE_VER                 = 83, //PE verification failed
AUTO_PE_NO_IN_ENV           = 84, //PE does not present in environment
AUTO_PE_CALIB               = 85, //invalid calibration/diagnostic data
AUTO_PC_DRV_STA_CONFLICT    = 86, //conflict between PC-driven and standalone modes
AUTO_CALIB_WORD_1_CORRUPT   = 87, //Calibration word 1 corrupted during programming
AUTO_CALIB_WORD_2_CORRUPT   = 88, //Calibration word 2 corrupted during programming
AUTO_ENV_NUM_OUT_RANGE      = 89, //Specified environment number is out of range
AUTO_DEMO_ERR               = 101}; //demo version
```

34 Appendix C: DEBUG and COE Bits

A compiler may provide incorrect settings for debug-related bits: DEBUG (Background debug) and/or COE (Clip-on emulation mode). Development tool as MPLAB IDE manipulate these bits automatically while ICP2 programmers load them from a HEX file "as is".

The following procedure is strongly recommended to validate these bits:

- Run MPLAB IDE
- Compile in "Release" mode
- Export the HEX file (File→Export) under a name (for example) "1.hex"
- Import HEX file "1.hex" (File→Import")
- Write down the checksum

- Run "ICP for Windows"
- Open "1.hex"
- Compare checksums
- Inspect configuration bits. If DEBUG or COE are set to debug mode then change them to operational one
- Save an updated buffers: File→Save As(Export)…

35 Revision History

- Revision 4.13.1a (Jan-13):
  - changed ICP software setup destination (new: C\Softlog\..., old: C:\Program Files\Soft-Log)
  - added ICP2-COMBO related info
  - added checksum calculation explanation
  - added LAN descriptiona
- Revision 4.12.1 (Aug-13):
  - added warning for serialization dialog – see 16.1
  - corrected example 16.2
  - added warning for ICP2-Portable – see 30.2
- Revision 4.10.2 (Aug-12):
  - added "Speed Optimization Utility" paragraph
  - added "Gap Eliminator™" paragraph
  - added new Preferences – see 19
- Revision 4.9.2 (Apr-2012):
  - added description of Enhanced ICSP limitations – see 19.7
  - changed appearance of "Preferences" – see 19
- Revision 4.9.1 (Jan-2012):
  - added description of "Enhanced ICSP™"
  - GO/PASS/FAIL outputs specified as optional for ICP2-Portable
- Revision 4.8.2 (Aug-2011):
  - added description of "raw" serialization method, serialization screenshots updated
- Revision 4.8.1 (Jul-2011):
  - added Preference "Automatically select PM (flash) range"
  - added standalone serialization – see 16.6
  - added paragraph "Viewing Environment" - see 15
  - added paragraph "Debug and COE Bits" - see 34

36 Technical Assistance

You may contact Softlog Systems for technical assistance by calling, sending a fax or e-mail. To help us give you quick and accurate assistance, please provide the following information:

- Software version number, firmware version number and product serial number (if available). This information is displayed at the program start
- Detailed description of the problem you are experiencing
- Error messages (if any)
- Microcontroller part number (if device-related)

- Send us your "icp01.cfg" file

37 Warranty

Softlog Systems (2006) Ltd. warrants this product against defects in materials and workmanship for a period of 1 (one) year. This warranty will not cover programmers that, in the opinion of Softlog Systems, have been damaged due to abuse, improper use, disassembly, replacement of parts or attempted repair by anyone other than an authorized Softlog Systems service technician.

This product must be returned to the supplier for warranty service within the stated period. The buyer shall pay all shipping costs and other charges or assessments for the product by the supplier.

Softlog Systems shall not be liable for any indirect, incidental, or consequential damages, regardless of whether liability is based upon breach of warranty, negligence, strict liability in tort, or any other theory, Softlog Systems will never be liable in an amount greater than the purchase price of the products described by this express warranty. No agent, distributor, salesperson, or wholesale or retail dealer has the authority to bind Softlog Systems to any other affirmation, representation, or warranty concerning these goods.

38 Contact

Softlog Systems (2006) Ltd.

6 Hayotzrim St. Or-Yehuda 6021820 Israel

Phone: 972-3-9515359
Fax: 972-3-9527520
Web: www.softlog.com
E-mail: sales@softlog.com, support@softlog.com

39 Copyright Notice