IP Lock

User's Manual

Design Gateway Co.,Ltd.

Rev 1.8

(PD0601-6-01-08E)

*** Please read this manual carefully before using IP Lock ***

Revision History

Revision	Date	Detail of change
1.0	10-May-06	Initial Release
1.1	01-Aug-06	Adding IP lock core for Altera FPGA.
1.2	14-Nov-06	Update detail of setting internal pull-up on ISE Update detail of SC0 signal
1.3	08-Nov-07	Support Xilinx Virtex5 Adding Troubleshooting
1.4	30-Dec-09	Update resource usage on Xilinx
1.5	06-Aug-10	Update Figure 4-1 and 4-8
1.6	15-Oct-10	Update Device support
1.7	18-Dec-18	Update How to install Device driver topic for window10
1.8	28-May-19	Add software installation

Table of Contents

1. Product Overview

1.1. Delivery items

Thank you very much for purchasing IP Lock. The product box includes the following items.

1. IP Lock writer : 1 piece
2. IP Lock device : 3 pieces
3. USB A-B cable : 1 piece
4. CD ROM : 1 piece
    - IP Lock Writer software (setup.exe)
    - IP Lock core (TopIPLock.vhd, iplock.vhd)
    - Example VHDL design source code (Counter.vhd, Counter32Bits.vhd)
    - IP Lock Writer user's manual (IPLWR_UserManualx_x_E.pdf)

Note: License file is necessary to implement IP Lock core into the user's project file by using FPGA tools (Vivado, ISE, or Quartus). The details to send the request for IP Lock license file are described in topic "2.2 The license file".

1.2. IP Lock core specification

1. 128-bit AES encryption
2. Resource

    Xilinx FPGA (Spartan-7 device) : 318 Slices,1 BRAM Tiles (36 Kb)

    Intel FPGA (Cyclone10 LP device) : 1221 LEs, 24,576 Memory bits

    ** Actual logic resource depends on the percentage of unrelated logic
3. Sampling period to check connection status between IP Lock device and IP Lock core:

    Every 200 msec
4. Supported device list

    Please download from https://dgway.com/products/IPLock/IPL-LIST.pdf.

1.3. System Requirement for IP Lock core

1. Xilinx FPGA tool

   a) ISE Design Suite for Spartan6, Virtex6 and the previous generation families

   Please see more details about the system requirement for ISE tool from following link.

   https://www.xilinx.com/products/design-tools/ise-design-suite.html

   https://www.xilinx.com/products/design-tools/ise-design-suite/memory.html

   b) Vivado Design Suite for Virtex7, Kintex7, Artix7, Zynq7000 and the newer families.

   Please see more details about the system requirement for Vivado tool from following link.

   https://www.xilinx.com/products/design-tools/vivado.html

   https://www.xilinx.com/products/design-tools/vivado/memory.html

2. Intel FPGA tool: Quartus tool

   Please check more details about the tools and the system requirement from following link.

   https://www.intel.com/content/www/us/en/programmable/support/support-resources/download/os-support.html

1.4. IP Lock caution

Please confirm the below information when using the IP Lock system.

1. Please write user's ID to IP Lock device by IP Lock writer before usage. IP Lock in the product box still not be programmed the user's ID.
2. IP Lock core and IP Lock writer must be used from the same product box. Each IP Lock writer has the unique writer's ID. So, the stuffs from different product box cannot communicate successfully.
3. Please check IP Lock device direction before mounting on the Writer and the PCB.
4. Support voltage range of IP Lock device: +1.8V, +2.5V or +3.3V.

1.5. Warranty Policy

1. Product warranty is valid for 1 year from purchasing date.
2. Warranty will be void if the following conditions are found.
   (a) Some modification has been made to this product.
   (b) The product is operated incorrectly (not follow the instruction in this manual).
   (c) The warranty sticker is torn or damaged.

1.6. Support

The support by email about IP Lock problem can be sent to iplock@design-gateway.com. Otherwise, the request on our website is available by following link.

http://www.design-gateway.com/contact.html

Your personal information is restricted with high confidentiality.

2. IP Lock core

2.1. The core operation

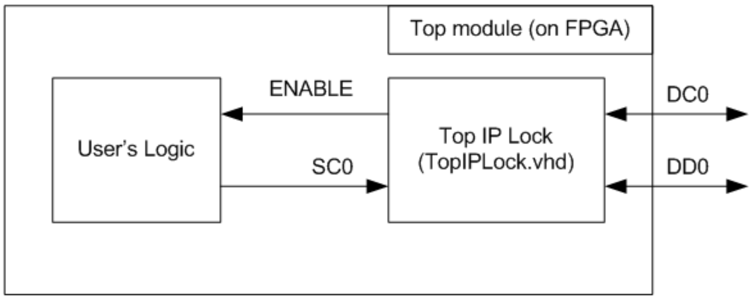


Figure 2-1 IP Lock System

As shown in Figure 2-1, the IP Lock core has four signals for connecting to other modules, i.e. SC0, DC0, DD0, and ENABLE. SC0 is the system clock for running IP Lock core. *The frequency range of SC0 is 1 – 25 MHz*. The DC0 and DD0 are the external I/Os for connecting with IP Lock device. Both signals are bi-directional signals. ENABLE signal is the output of IP Lock core for the user's logic. When the link communication between IP Lock core and IP Lock device is successfully created, ENABLE signal will be asserted to '1'. Otherwise, ENABLE signal is de-asserted to '0' (the communication is failed).

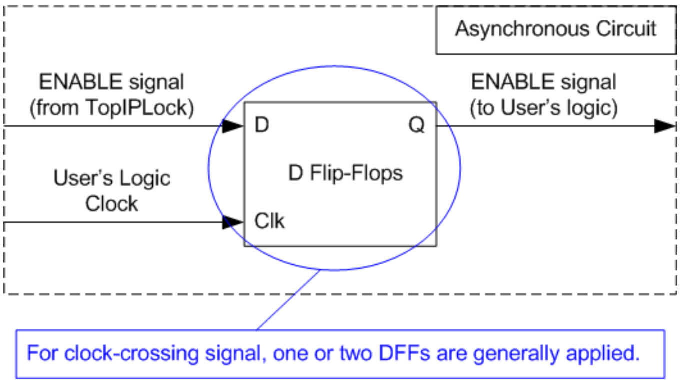


Figure 2-2 Recommended logic for using ENABLE signal

The user's logic can use ENABLE signal to enable the internal operation by monitoring ENABLE to be equal to '1'. It's recommended for the user's logic to add one or two DFFs before using ENABLE signal to be the synchronous circuit, as shown in Figure 2-2.

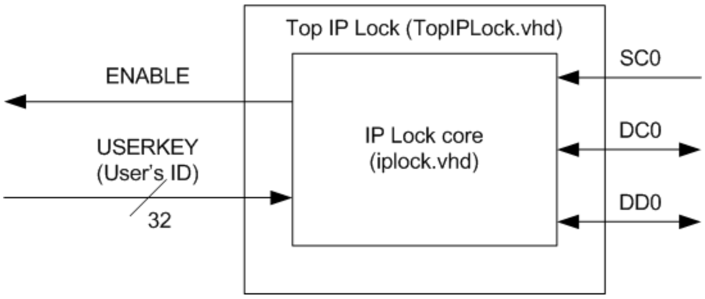
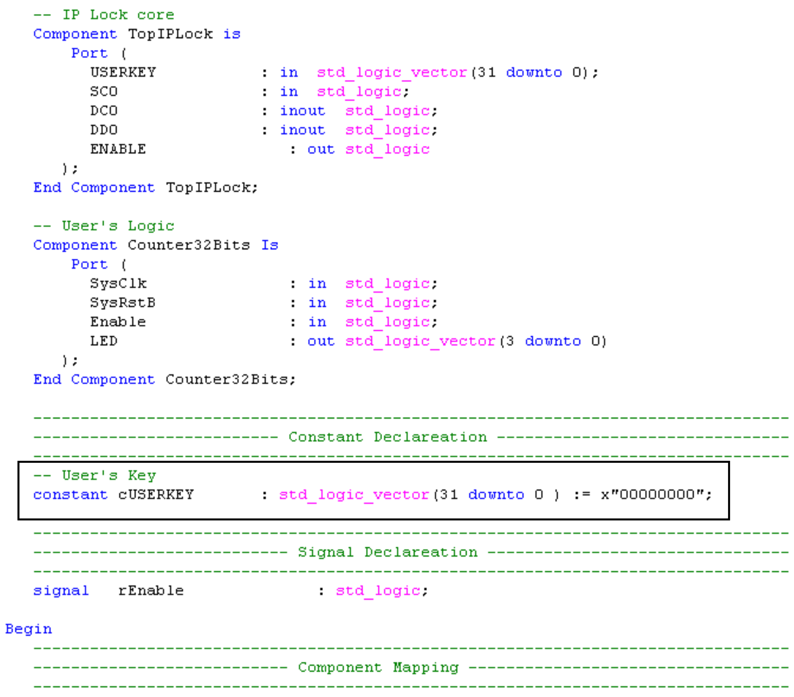


Figure 2-3 Top level of IP Lock

TopIPLock (TopIPLock.vhd) is designed to be the wrapper file of IP Lock core (iplock.vhd) to show the input/output signals of the core. USERKEY which is input signal to IP Lock core is the user's ID which must be set to the same value as the user's ID within the IP Lock device which is set by the IP Lock Writer software. If the user's ID on HDL logic does not match to the user's ID on IP Lock device, the ENABLE signal will be de-asserted to '0'.



```
    -- IP Lock core
    Component TopIPLock is
        Port (
          USERKEY          : in  std_logic_vector(31 downto 0);
          SC0              : in  std_logic;
          DC0              : inout  std_logic;
          DD0              : inout  std_logic;
          ENABLE              : out std_logic
        );
    End Component TopIPLock;

    -- User's Logic
    Component Counter32Bits Is
        Port (
          SysClk              : in  std_logic;
          SysRstB             : in  std_logic;
          Enable              : in  std_logic;
          LED                 : out std_logic_vector(3 downto 0)
        );
    End Component Counter32Bits;

    --------------------------------------------------------------------------------
    -------------------------- Constant Declareation ------------------------------
    --------------------------------------------------------------------------------
    -- User's Key
    constant cUSERKEY       : std_logic_vector(31 downto 0 ) := x"00000000";
    --------------------------------------------------------------------------------
    --------------------------- Signal Declareation -------------------------------
    --------------------------------------------------------------------------------
    signal   rEnable              : std_logic;

Begin
    --------------------------------------------------------------------------------
    --------------------------- Component Mapping ---------------------------------
    --------------------------------------------------------------------------------
```

Figure 2-4 User's ID in source code

In the example HDL code, USERKEY is assigned by 32 bit constant value, as shown in Figure 2-4. To change the user's ID, the user updates value in the HDL and then updates the same value to the IP Lock device by using the Software.

2.2. The license file

To implement the IP Lock core by using FPGA tool (ISE/Vivado for Xilinx device or Quartus for Intel device), the user must send email to iplock@design-gateway.com for generating the IP Lock license file. The information in Figure 2-5 must be sent to email for generating the license file. The user can select to lock the license file by using Volume ID or Network ID.

Name / Company: ……………………………………………………..

IP Lock Serial Number: ……………………………………………….

Volume / Network ID: ………………….....................................

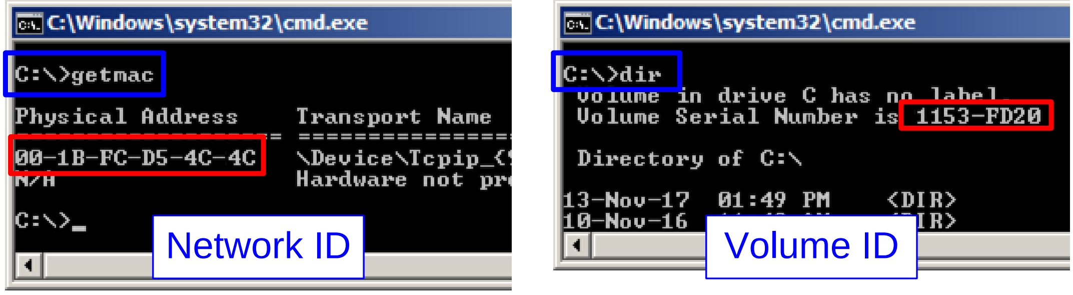
Address: ………………………………………………………………..

Tel: ……………………………………………………………………….

Fax: …………………………………………………………….....

Figure 2-5 Information for register

Figure 2-6 shows the command to check Volume ID and Network ID by using Windows command prompt. “getmac” is the command to check Network ID while “dir” is applied to check Volume ID.



```
C:\>getmac

Physical Address    Transport Name
=================== ===============
00-1B-FC-D5-4C-4C   \Device\Tcpip_{
N/A                 Hardware not pr

C:\>_
```

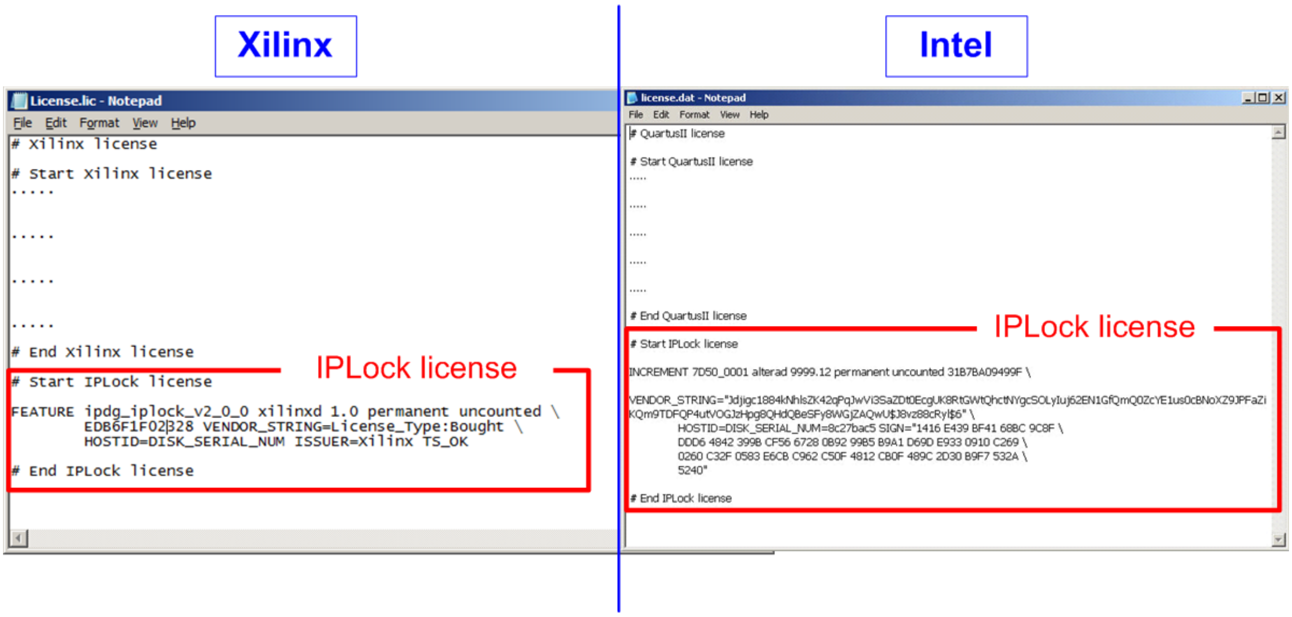
```
C:\>dir
 Volume in drive C has no label.
 Volume Serial Number is 1153-FD20

 Directory of C:\

13-Nov-17  01:49 PM    <DIR>
10-Nov-16                <DIR>
```

Figure 2-6 Volume Serial Number

After the user gets the license file from Design Gateway, the user must add the license file to the tool license by using text editor. Assume that, “License.lic” is the license file of Xilinx tool (ISE or Vivado) while “License.dat” is the license file of Intel tool (Quartus). Figure 2-7 shows the example to add IP Lock license to Xilinx and Intel tool.



```
# Xilinx license

# Start Xilinx license
.....

.....

.....

.....
# End Xilinx license

# Start IPLock license

FEATURE ipdg_iplock_v2_0_0 xilinxd 1.0 permanent uncounted \
        EDB6F1F02328 VENDOR_STRING=License_Type:Bought \
        HOSTID=DISK_SERIAL_NUM ISSUER=Xilinx TS_OK

# End IPLock license
```

```
# QuartusII license

# Start QuartusII license
.....

.....

.....

.....

.....

# End QuartusII license

# Start IPLock license

INCREMENT 7D50_0001 alterad 9999.12 permanent uncounted 31B7BA09499F \

VENDOR_STRING="Jdjigc1884kNhlsZK42qPqJwVi3SaZDt0EcgUK8RtGWtQhctNYgcSOLyIuj62EN1GfQmQ0ZcYE1us0cBNoXZ9JPFaZi
KQm9TDFQP4utVOGJzHpg8QHdQBeSFy8WGjZAQwU$J8vz88cRyl$6" \
        HOSTID=DISK_SERIAL_NUM=8c27bac5 SIGN="1416 E439 BF41 68BC 9C8F \
        DDD6 4842 399B CF56 6728 0B92 99B5 B9A1 D69D E933 0910 C269 \
        0260 C32F 0583 E6CB C962 C50F 4812 CB0F 489C 2D30 B9F7 532A \
        5240"

# End IPLock license
```

Figure 2-7 Add IP Lock license to the tool's license

2.3. The example VHDL design

The example VHDL code to use IP Lock core is included in the delivery item. The hardware block diagram in the example design is shown in Figure 2-8.

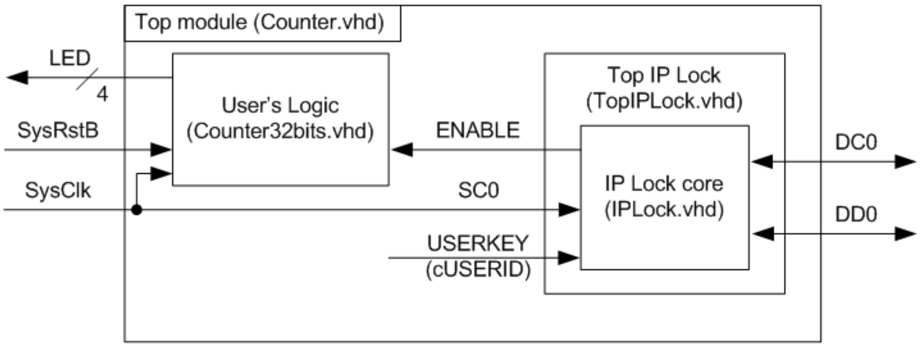
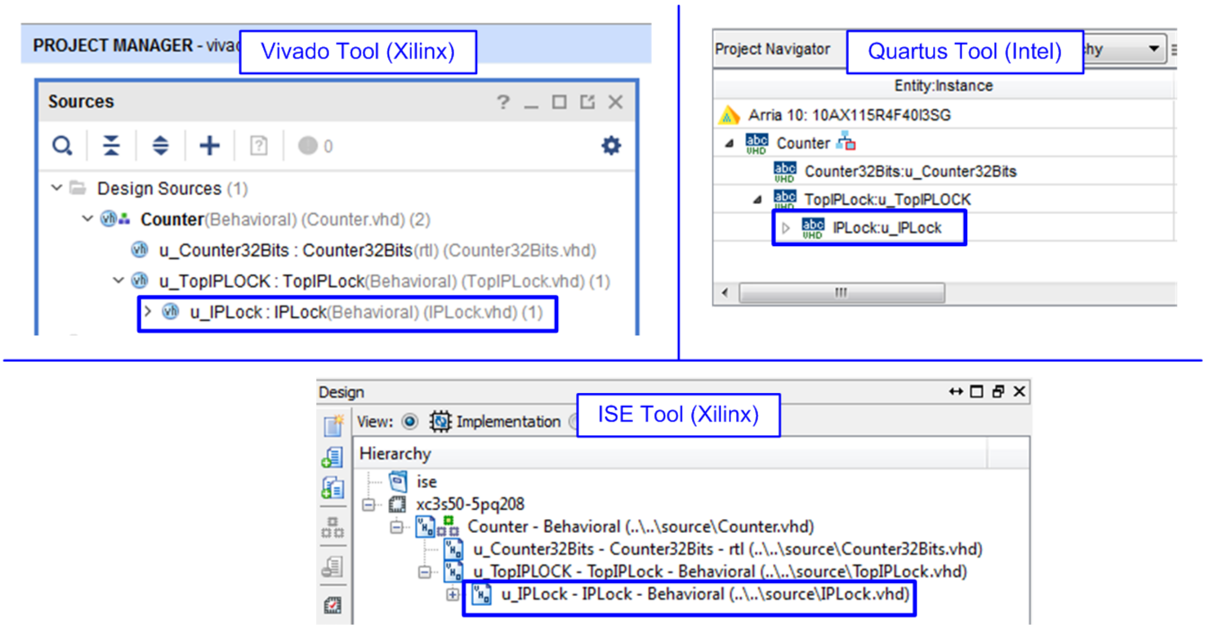


Figure 2-8 Example VHDL design block diagram

Counter.vhd is the top module which includes the example of user's logic and the IP lock core. The example of user's logic is Counter32bits.vhd. The user's logic uses ENABLE signal to be counter enable to blink LED[0] and LED[1] signal. SysClk is applied to be clock input for user's logic and IP Lock core.

The example project to use IP Lock core (IPLock.vhd) in each FPGA tool is shown in Figure 2-9. The IP Lock core is added to the project as the source like other HDL modules.



Design Sources (1)
Counter(Behavioral) (Counter.vhd) (2)
u_Counter32Bits : Counter32Bits(rtl) (Counter32Bits.vhd)
u_TopIPLOCK : TopIPLock(Behavioral) (TopIPLock.vhd) (1)
u_IPLock : IPLock(Behavioral) (IPLock.vhd) (1)

Entity:Instance
Arria 10: 10AX115R4F40I3SG
Counter
Counter32Bits:u_Counter32Bits
TopIPLock:u_TopIPLOCK
IPLock:u_IPLock

Hierarchy
ise
xc3s50-5pq208
Counter - Behavioral (..\..\source\Counter.vhd)
u_Counter32Bits - Counter32Bits - rtl (..\..\source\Counter32Bits.vhd)
u_TopIPLOCK - TopIPLock - Behavioral (..\..\source\TopIPLock.vhd)
u_IPLock - IPLock - Behavioral (..\..\source\IPLock.vhd)

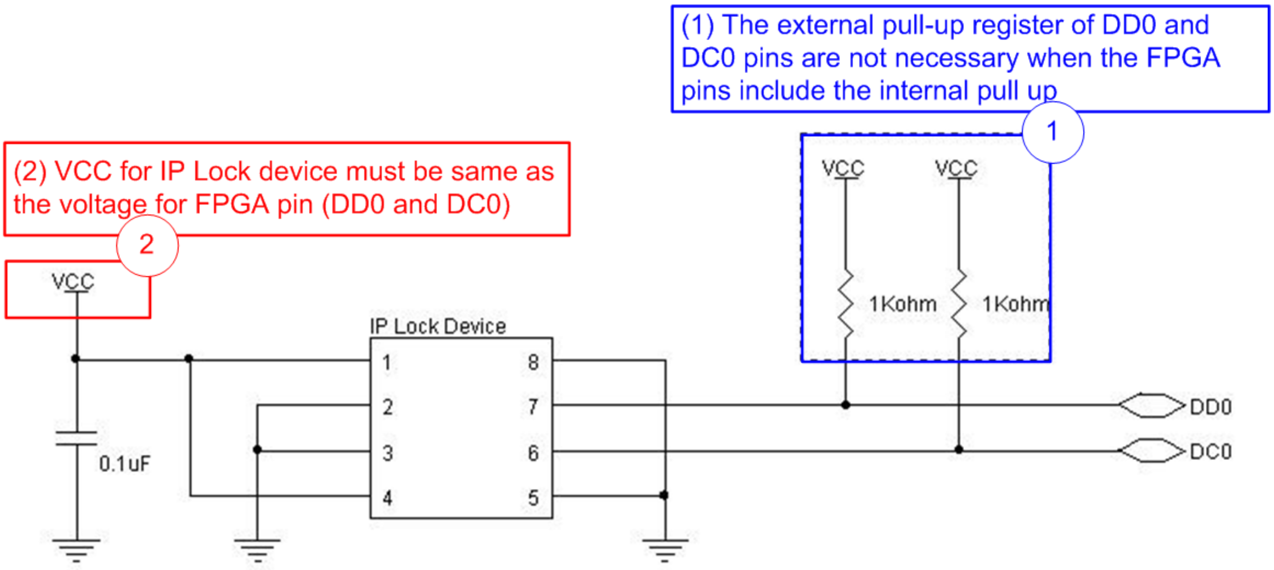
Figure 2-9 Example project for IP Lock core in each FPGA tool

3. IP Lock Hardware

3.1. IP Lock Device

IP Lock device is designed to communicate with the IP Lock core (implemented within FPGA) to protect FPGA core. The connection between IP Lock device and IP Lock core must be online status. If the communication link is lost, the IP Lock core will change the status of ENABLE signal to '0' to disable the user's logic inside the FPGA.

The hardware connections to use the IP lock device are shown in Figure 3-1 and Figure 3-2. Voltage of IP Lock device and FPGA I/O Voltage for connecting to IP Lock device must be fed from the same source. The supported voltages are +1.8V, +2.5V, and +3.3V.



(1) The external pull-up register of DD0 and DC0 pins are not necessary when the FPGA pins include the internal pull up
1
(2) VCC for IP Lock device must be same as the voltage for FPGA pin (DD0 and DC0)
2
VCC
VCC
VCC
1Kohm
1Kohm
IP Lock Device
1
2
3
4
8
7
6
5
0.1uF
DD0
DC0

Figure 3-1 IP Lock device connection when internal pull up is not included in FPGA pin

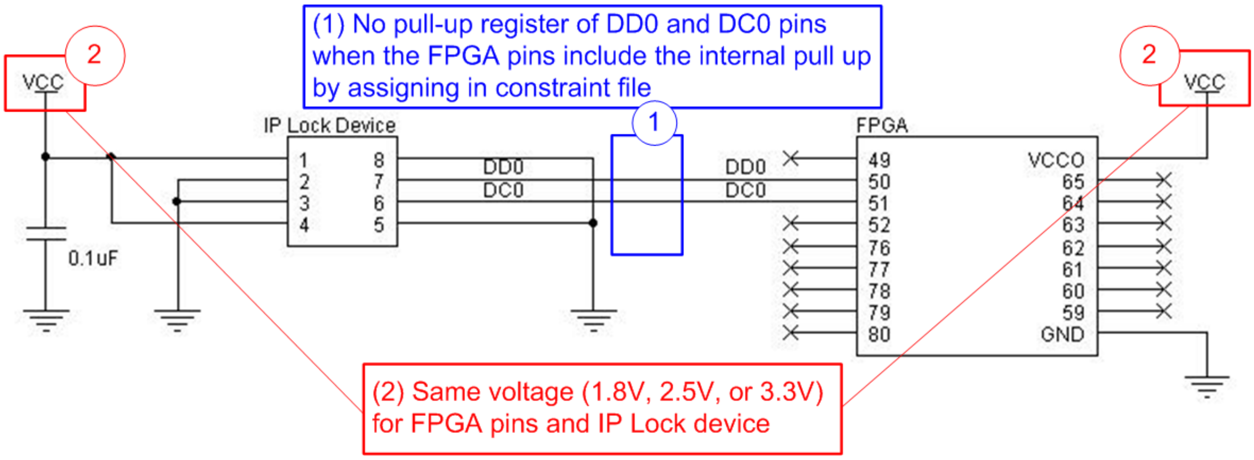
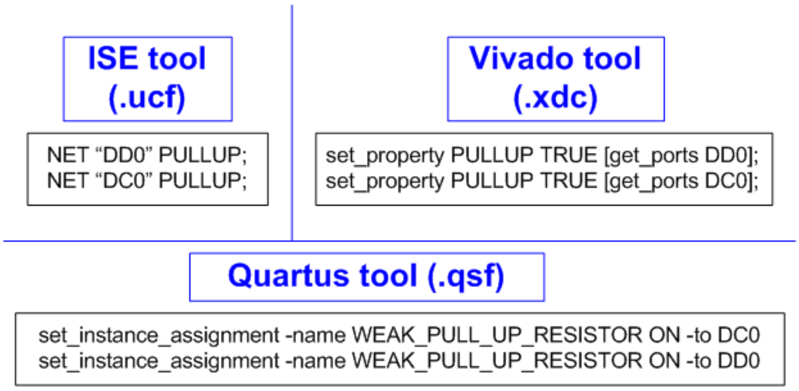

(1) No pull-up register of DD0 and DC0 pins when the FPGA pins include the internal pull up by assigning in constraint file
1
2
VCC
2
VCC
IP Lock Device
1
2
3
4
8
7
6
5
DD0
DC0
DD0
DC0
0.1uF
FPGA
49
50
51
52
76
77
78
79
80
VCCO
65
64
63
62
61
60
59
GND
(2) Same voltage (1.8V, 2.5V, or 3.3V) for FPGA pins and IP Lock device

Figure 3-2 IP Lock device connection when internal pull up is included in FPGA pin

The example constraint file to add pull up register for FPGA pin is shown in Figure 3-3. Three examples are shown for three FPGA tools, i.e. ucf file for ISE tool (Xilinx), xdc file for Vivado tool (Xilinx), and qsf file for Quartus tool (Intel).


ISE tool (.ucf)

```
NET "DD0" PULLUP;
NET "DC0" PULLUP;
```

Vivado tool (.xdc)

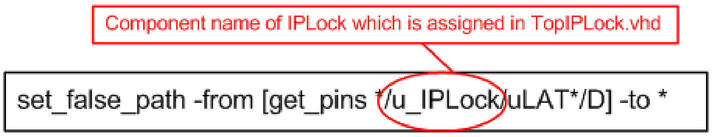
```
set_property PULLUP TRUE [get_ports DD0];
set_property PULLUP TRUE [get_ports DC0];
```

Quartus tool (.qsf)

```
set_instance_assignment -name WEAK_PULL_UP_RESISTOR ON -to DC0
set_instance_assignment -name WEAK_PULL_UP_RESISTOR ON -to DD0
```

Figure 3-3 Add pull up register to FPGA pin

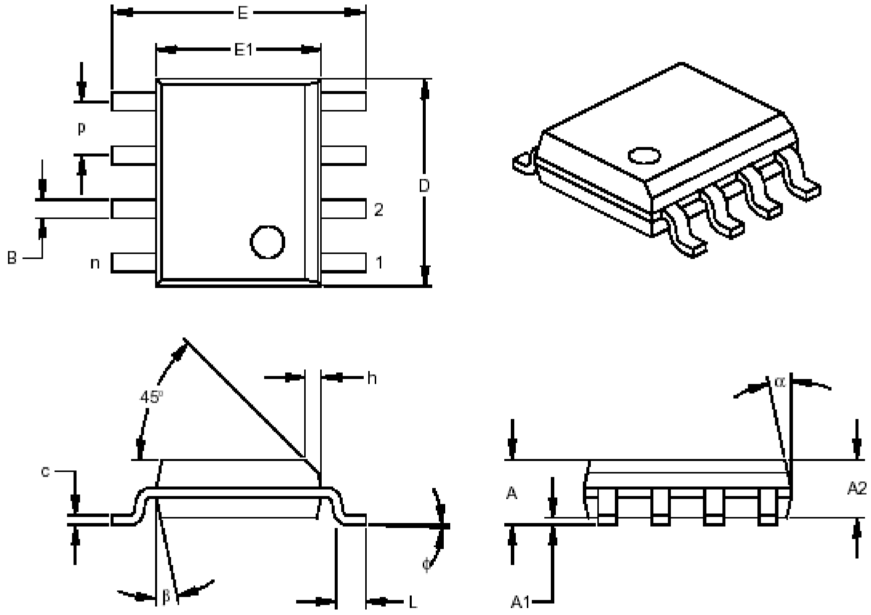
Note: When the IP Lock core is implemented by Vivado tool, the user must set the additional constraint file to ignore timing constraint for the internal signal of IP Lock core, as shown in Figure 3-4. "u_IPLock" in the example is the component name of IP Lock core, set in TopIPLock.vhd. If the user modifies the component name of IP Lock core within TopIPLock.vhd, the ignore path must be modified to match the new component name.


Component name of IPLock which is assigned in TopIPLock.vhd

```
set_false_path -from [get_pins */u_IPLock/uLAT*/D] -to *
```

Figure 3-4 Ignore path for Vivado tool

The IP Lock device packages and dimensions are shown in Figure 3-5 and Figure 3-6.



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n	8			8		
Pitch	p		.050			1.27	
Overall Height	A	.053	.061	.069	1.35	1.55	1.75
Molded Package Thickness	A2	.052	.056	.061	1.32	1.42	1.55
Standoff §	A1	.004	.007	.010	0.10	0.18	0.25
Overall Width	E	.228	.237	.244	5.79	6.02	6.20
Molded Package Width	E1	.146	.154	.157	3.71	3.91	3.99
Overall Length	D	.189	.193	.197	4.80	4.90	5.00
Chamfer Distance	h	.010	.015	.020	0.25	0.38	0.51
Foot Length	L	.019	.025	.030	0.48	0.62	0.76
Foot Angle	ϕ	0	4	8	0	4	8
Lead Thickness	c	.008	.009	.010	0.20	0.23	0.25
Lead Width	B	.013	.017	.020	0.33	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

* Controlling Parameter

§ Significant Characteristic

Figure 3-5 Package dimension of IP Lock device

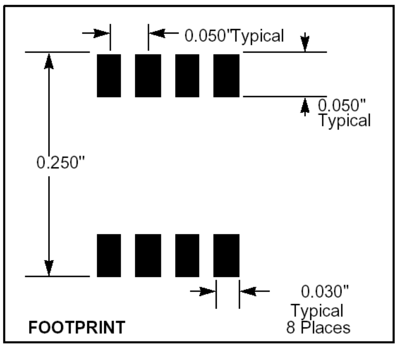


Figure 3-6 Footprint of IP Lock device (all dimensions are in inch unit)

3.2. IP Lock Writer

The IP Lock writer is applied to program the user's ID to the IP Lock device by using IP Lock software. The writer uses the power from USB cable, so it does not need to connect the external power supply.

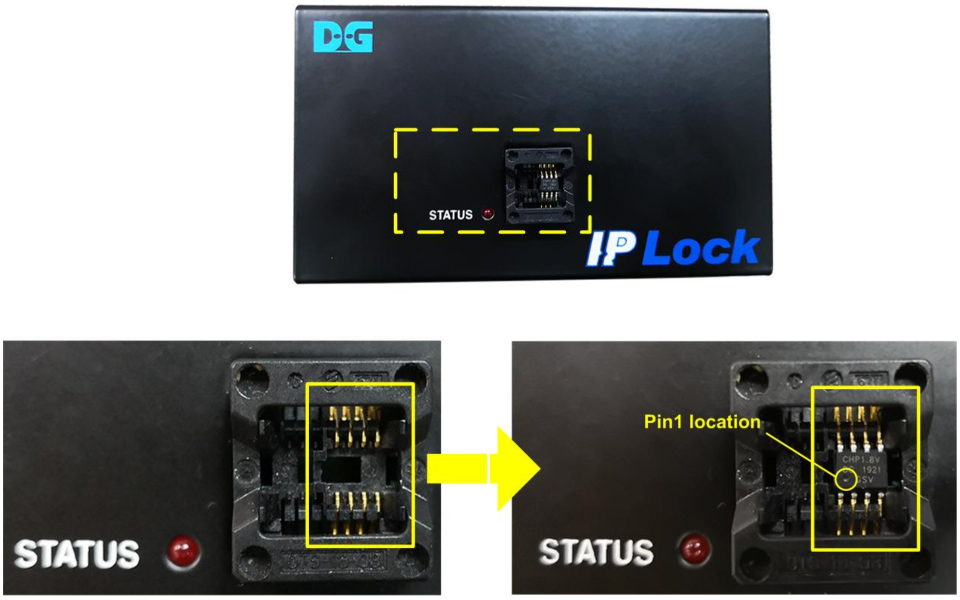


Figure 3-7 IP Lock writer

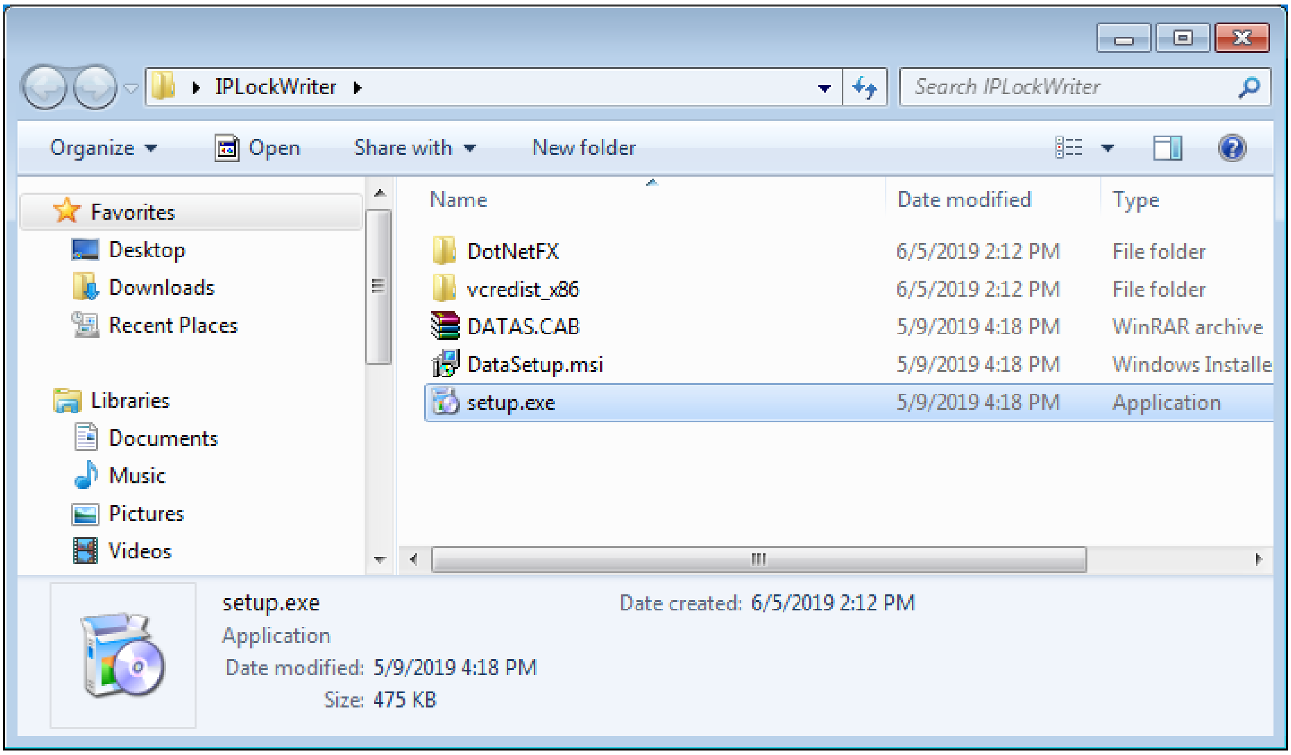
To place IP Lock device into the socket of IP Lock writer, please follow Figure 3-7. Pin 1 of IP Lock device must be placed at the same position as the status LED position.

4. IP Lock Writer Software

4.1. Software Installation

Please follow these steps to install the IP Lock Writer software.

1. Open CD from the package and select "setup.exe" to install the software.

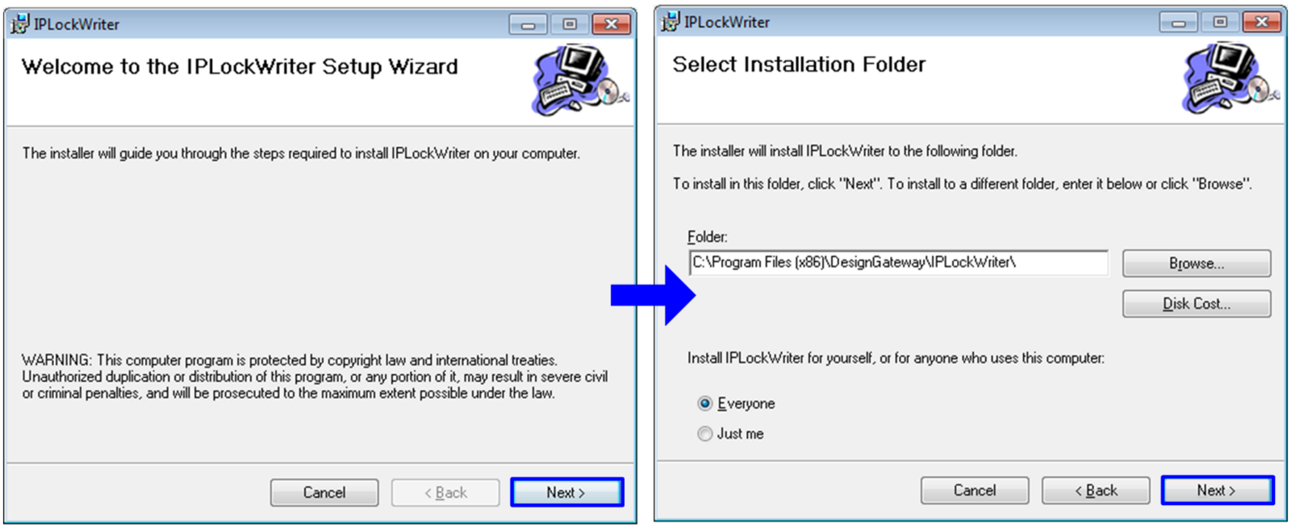


Name	Date modified	Type
DotNetFX	6/5/2019 2:12 PM	File folder
vcredist_x86	6/5/2019 2:12 PM	File folder
DATAS.CAB	5/9/2019 4:18 PM	WinRAR archive
DataSetup.msi	5/9/2019 4:18 PM	Windows Installe
setup.exe	5/9/2019 4:18 PM	Application

setup.exe
Application
Date modified: 5/9/2019 4:18 PM
Size: 475 KB
Date created: 6/5/2019 2:12 PM

Figure 4-1 Setup IP Lock Writer Software

2. Click "Next" button to start Setup wizard, and then click "Next" button to setup the installation folder.



Welcome to the IPLockWriter Setup Wizard

The installer will guide you through the steps required to install IPLockWriter on your computer.

WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.

Select Installation Folder

The installer will install IPLockWriter to the following folder.

To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".

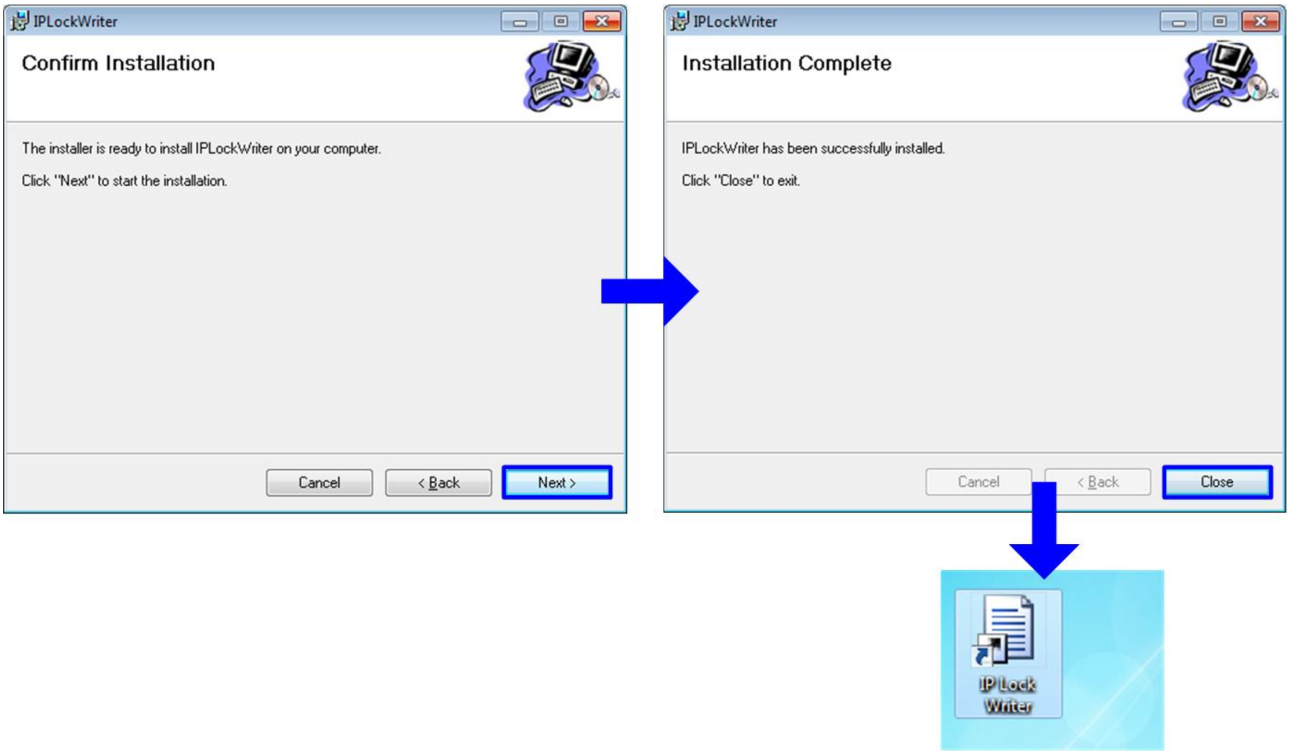
Folder: C:\Program Files (x86)\DesignGateway\IPLockWriter\

Install IPLockWriter for yourself, or for anyone who uses this computer:

● Everyone
○ Just me

Figure 4-2 Setup installation folder

3. Click "Next" button to confirm the installation. After the installation is finished, click "Close" button to complete the setup. Finally, IP Lock Writer icon is appeared on the desktop.



IPLockWriter

Confirm Installation

The installer is ready to install IPLockWriter on your computer.

Click "Next" to start the installation.

IPLockWriter

Installation Complete

IPLockWriter has been successfully installed.

Click "Close" to exit.

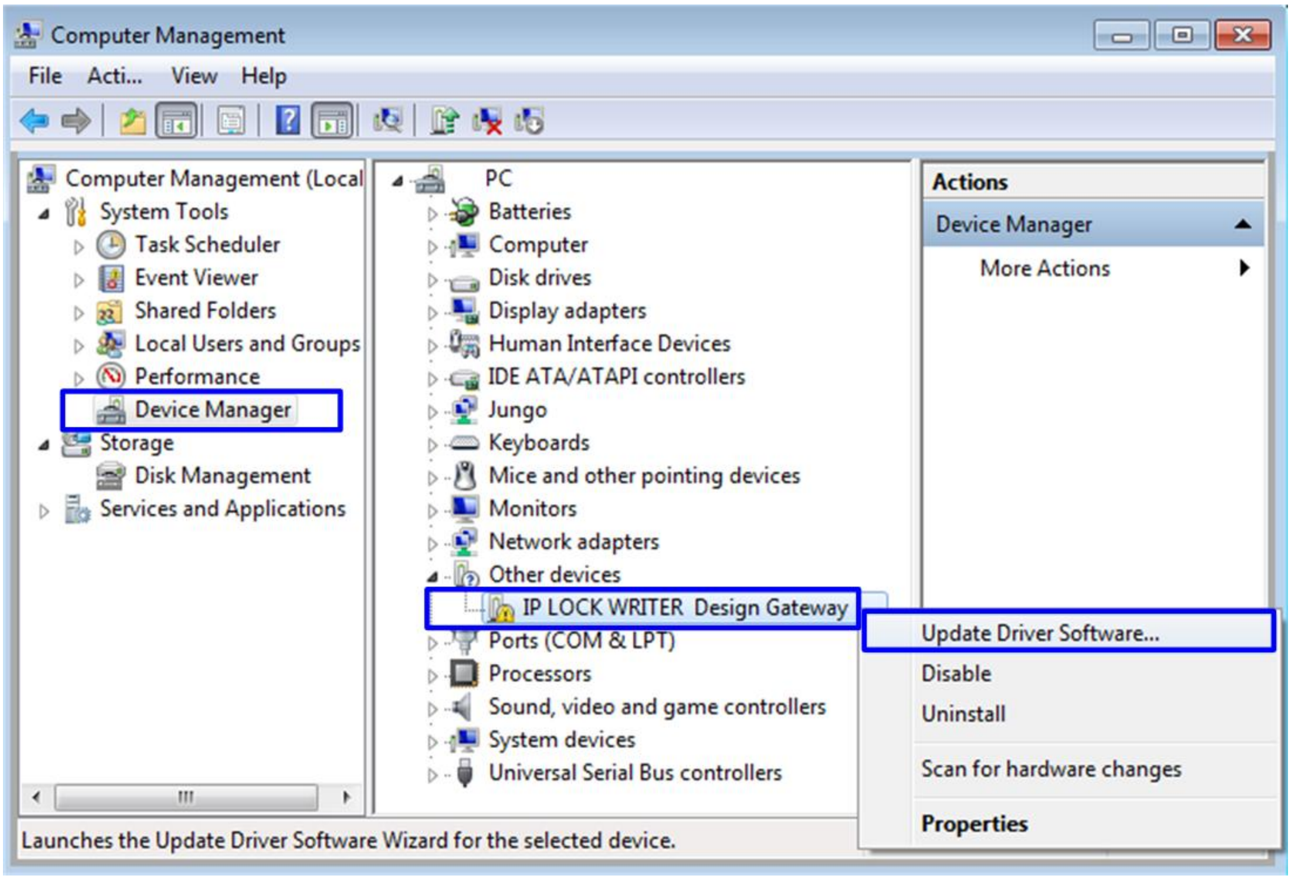
Figure 4-3 Installation complete

After finishing the IP Lock Writer installation, the user can plug in the IP Lock Writer to the PC. Next, the user needs to install the device driver of IP Lock Writer to the OS. The step to install the device driver is described in the next topic.

4.2. Device Driver Installation

Please follow these steps to install the device driver of IP Lock Writer.

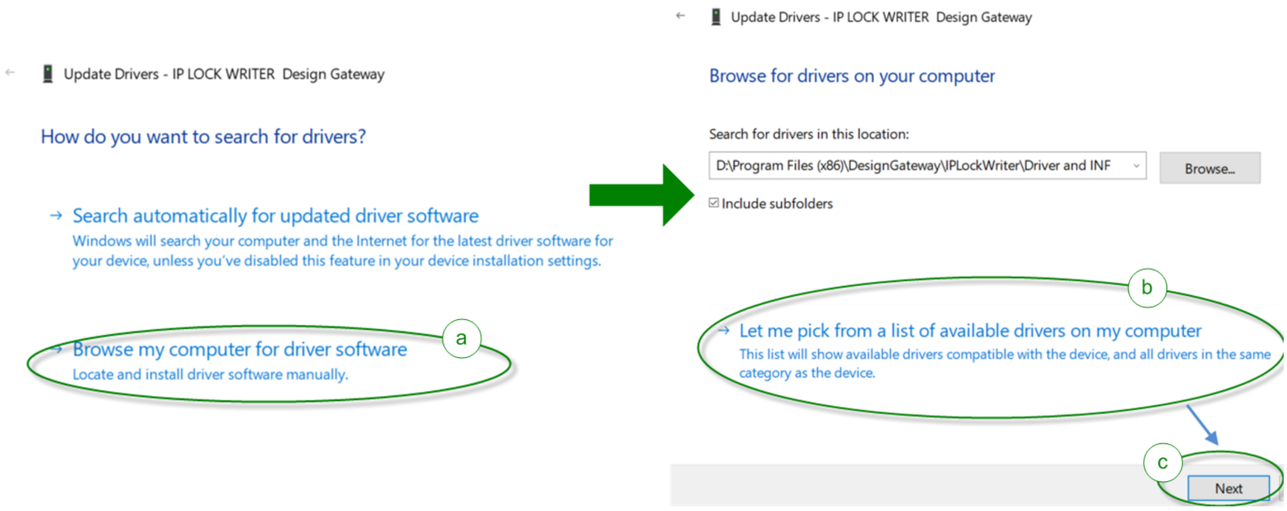
1. Plug in USB A-B cable from IP Lock Writer to USB connector on PC.
2. Open Device Manager of PC and the new device named “IP LOCK WRITER Design Gateway” is detected under “Other devices” category.
3. Right-Click at the device icon and then select “Update Driver Software…”.



Launches the Update Driver Software Wizard for the selected device.

Figure 4-4 IPLock Writer device detected

4. Select “Browse my computer for driver software” and then select “Let me pick from a list of available drivers on my computer”. Click “Next” button to continue the next step.



Update Drivers - IP LOCK WRITER Design Gateway

How do you want to search for drivers?

→ Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

→ Browse my computer for driver software
Locate and install driver software manually.

Update Drivers - IP LOCK WRITER Design Gateway

Browse for drivers on your computer

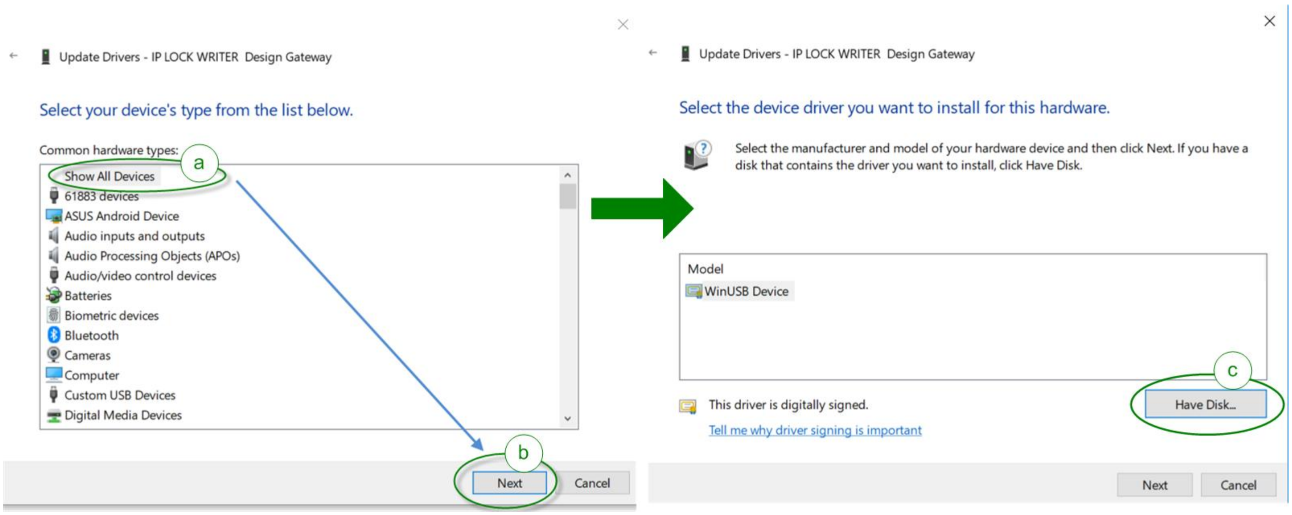
Search for drivers in this location:
D:\Program Files (x86)\DesignGateway\IPLockWriter\Driver and INF

☑ Include subfolders

→ Let me pick from a list of available drivers on my computer
This list will show available drivers compatible with the device, and all drivers in the same category as the device.

Figure 4-5 Browse the driver

5. Select “Show All Devices” and then click “Next” button. After that, click “Have Disk” button.



Update Drivers - IP LOCK WRITER Design Gateway

Select your device's type from the list below.

Common hardware types:
- Show All Devices
- 61883 devices
- ASUS Android Device
- Audio inputs and outputs
- Audio Processing Objects (APOs)
- Audio/video control devices
- Batteries
- Biometric devices
- Bluetooth
- Cameras
- Computer
- Custom USB Devices
- Digital Media Devices

Update Drivers - IP LOCK WRITER Design Gateway

Select the device driver you want to install for this hardware.

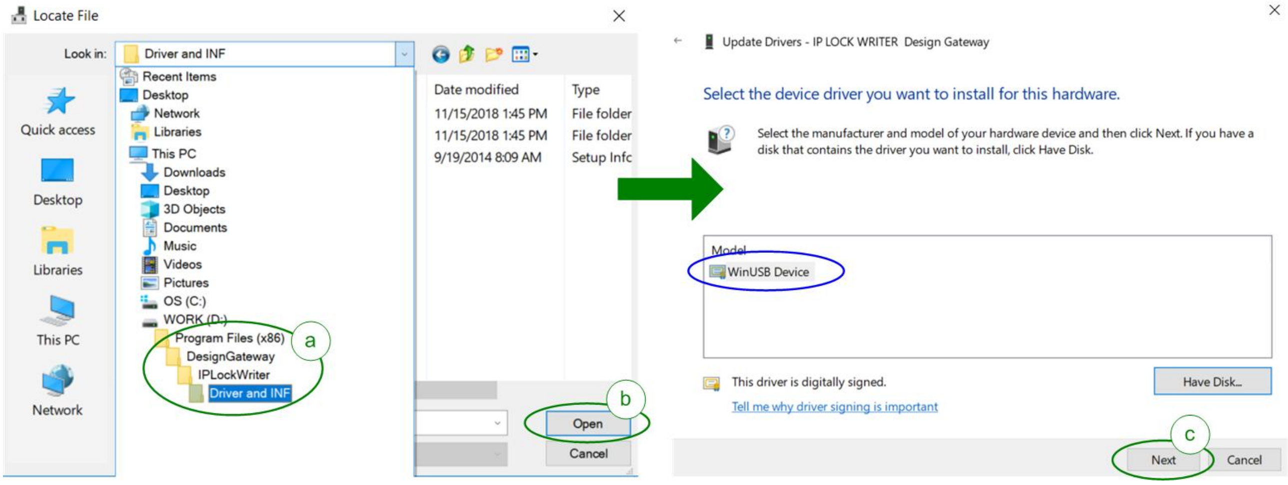
Select the manufacturer and model of your hardware device and then click Next. If you have a disk that contains the driver you want to install, click Have Disk.

Model
WinUSB Device

This driver is digitally signed.
Tell me why driver signing is important

Figure 4-6 Select device’s type from the list

6. Select INF file from \installation folder\DesignGateway\IPLockWriter\Driver and INF, and then click open button. After that, "WinUSB device" model is displayed. Click "Next" button to continue the next step.



Select the device driver you want to install for this hardware.

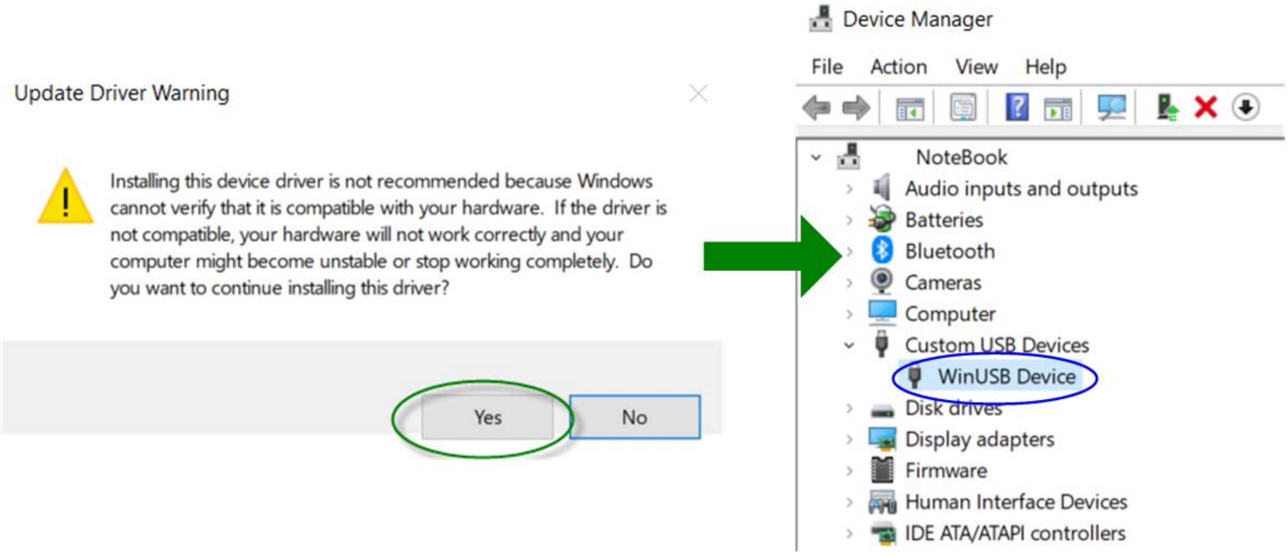
Select the manufacturer and model of your hardware device and then click Next. If you have a disk that contains the driver you want to install, click Have Disk.

This driver is digitally signed.

Tell me why driver signing is important

Figure 4-7 Choose INF file

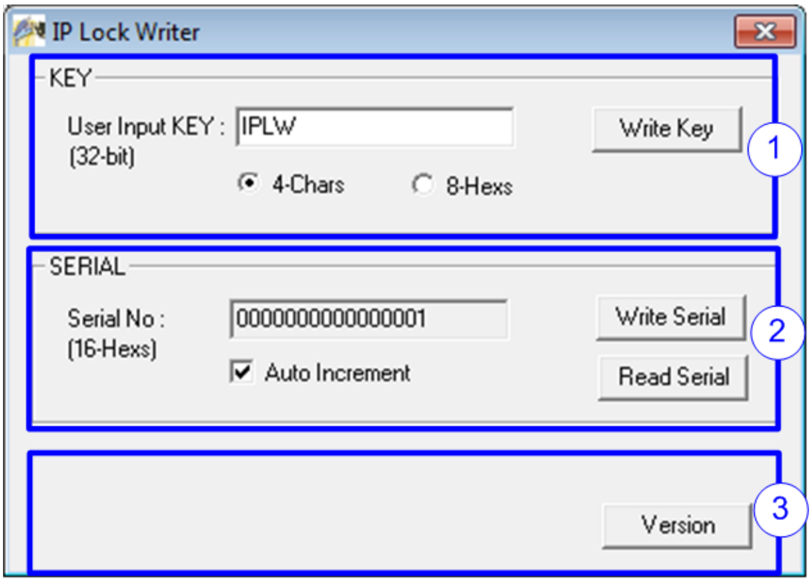
7. The warning dialog is displayed. Click "Yes" button to complete the installation. After finishing the installation, "WinUSB Device" is detected as Custom USB Devices.



Installing this device driver is not recommended because Windows cannot verify that it is compatible with your hardware. If the driver is not compatible, your hardware will not work correctly and your computer might become unstable or stop working completely. Do you want to continue installing this driver?

Figure 4-8 Device driver installation successful

4.3. Running Software



IP Lock Writer

KEY

User Input KEY : IPLW
(32-bit)

● 4-Chars ○ 8-Hexs

SERIAL

Serial No : 0000000000000001
(16-Hexs)

☑ Auto Increment

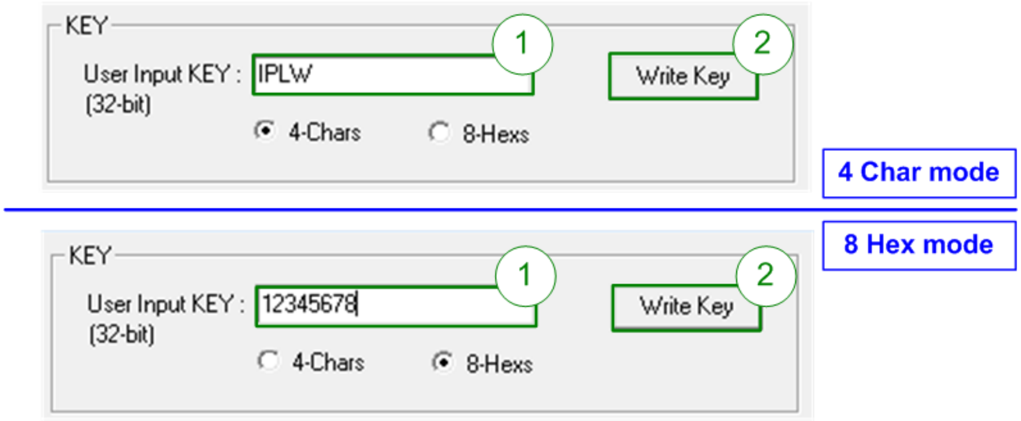
Figure 4-9 IP Lock Writer software

IP Lock writer software has three controls, i.e. Key, Serial, and Version.

1. The user input key is 32 bit value to be the user's ID for setting to the IP Lock device. This value must be matched to the user's ID setting in the HDL code. The user input key could be applied to refer the lot number of the IP Lock device. The factory default value of the user's ID is 0x00000000. The user can set the user's ID to IP Lock device only one time. Please confirm the set value before clicking "Write Key" button.
2. Serial number is 64 bit value to be the index of IP Lock device in each lot. Serial number can be set and read back by the software only. The Serial number is auto-written to IP Lock device with the user's ID when the user clicks "Write Key" button.
3. Version is applied to read the version of IP Lock writer firmware and IP Lock writer software.

More details to set Key control and Serial control are described as follows.

4.3.1 KEY

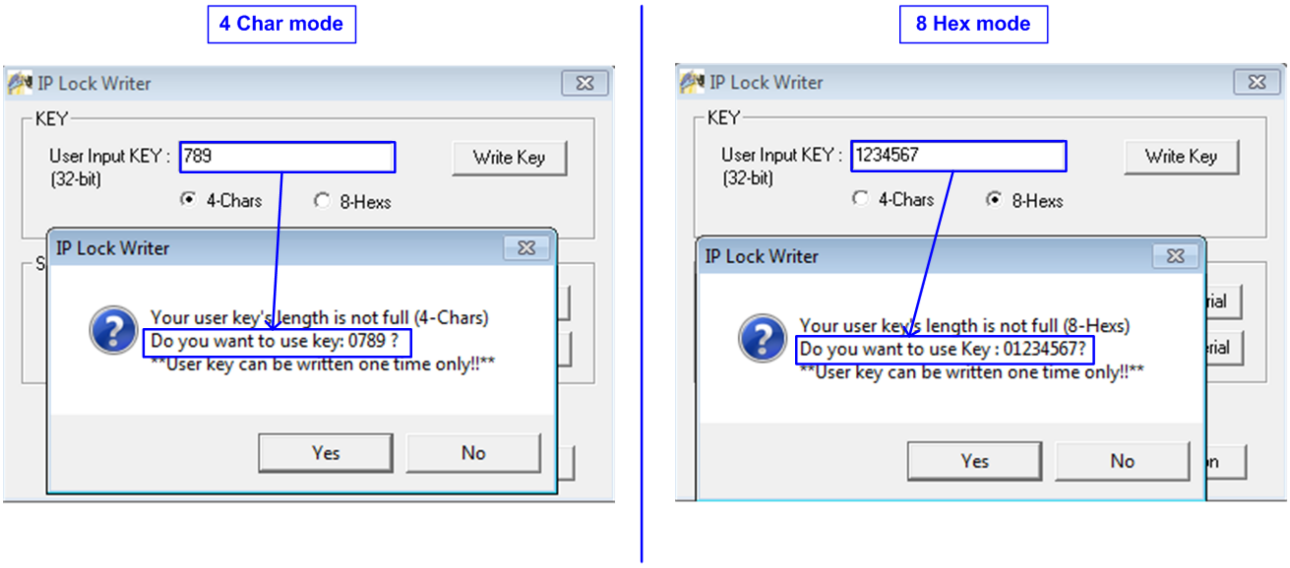


User Input KEY : IPLW
(32-bit)
● 4-Chars ○ 8-Hexs

User Input KEY : 12345678
(32-bit)
○ 4-Chars ● 8-Hexs

Figure 4-10 User Input Key

To input the user key, two input formats are supported, i.e. 4 characters and 8 hexadecimal digits. If the input is valid, the software will convert the user input key to 32 bit value (1 character is converted to 8 bit following the ASCII table and 1 hexadecimal digit is equal to 4 bit). When the user input is less than 4 characters or 8 hex digits, the software will fill ‘0’ character (0x30 as hex value) for char format or fill 0x00 for hex format.



4 Char mode:

User Input KEY : 789
(32-bit)
● 4-Chars ○ 8-Hexs

Your user key's length is not full (4-Chars)
Do you want to use key: 0789 ?
User key can be written one time only!!

8 Hex mode:

User Input KEY : 1234567
(32-bit)
○ 4-Chars ● 8-Hexs

Your user key's length is not full (8-Hexs)
Do you want to use Key : 01234567?
User key can be written one time only!!

Figure 4-11 Example when the number of input is too less

The examples to extend the input to 32 bit when the number of input is too less are as follows.

(a) If the user inputs “789” as char input, the software will fill the input to be “0789”. So, the user’s ID is equal to 0x30373839.

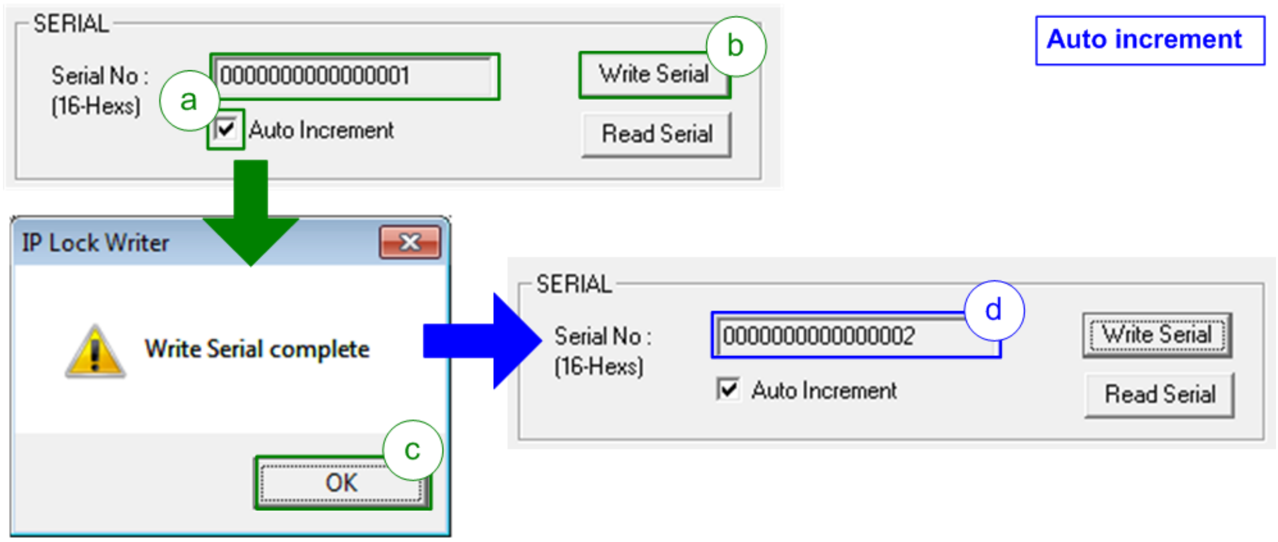
(b) If the user inputs “1234567” as hex input, the software will fill the input to be “01234567”. So, the user’s ID is equal to 0x01234567.

After the user clicks “Write Key” button, the software sets the user’s ID and the Serial Number to IP Lock device. The user’s ID could be set only one time for each IP Lock device. The user should check User input key value and Serial Number value before starting “Write Key” operation.

It is recommended the user to check “Auto increment” box. In this mode, Serial Number is auto increment after finishing setting Serial number to each IP Lock device. As a result, the user can set the same user’s ID but the different Serial Number to each IP Lock device by clicking “Write Key” button.

4.3.2 SERIAL

Serial control box is designed to set Serial number to IP Lock device and read back. To set Serial number, the user can run two modes.



Serial No : (16-Hexs) 0000000000000001

☑ Auto Increment

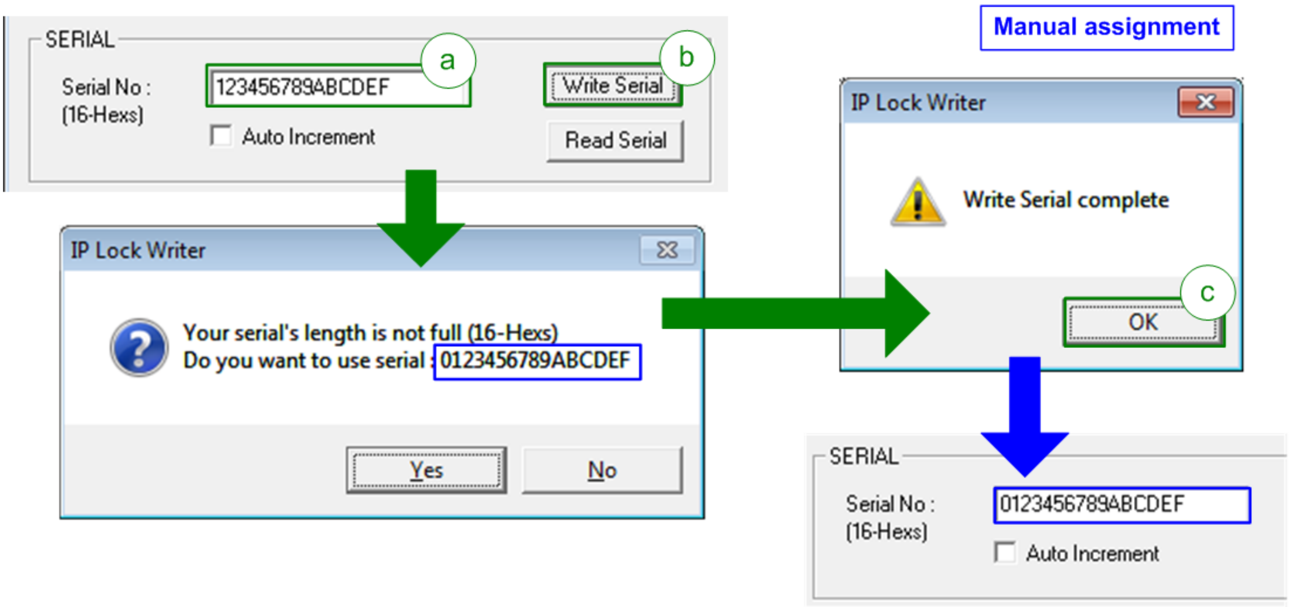
Write Serial complete

Serial No : (16-Hexs) 0000000000000002

☑ Auto Increment

Figure 4-12 Auto increment mode

First is Auto increment mode. In this mode, the software loads the start value from Serial No box to set to IP Lock device. When the user clicks “Write Serial” button, the software sets the value from the box to IP Lock device. “Write Serial complete” dialog is displayed after finishing the operation. After that, the value in Serial No box is increment as shown in Figure 4-12. If the user clicks “Write Serial” button again, the new value will be programmed to IP Lock device.

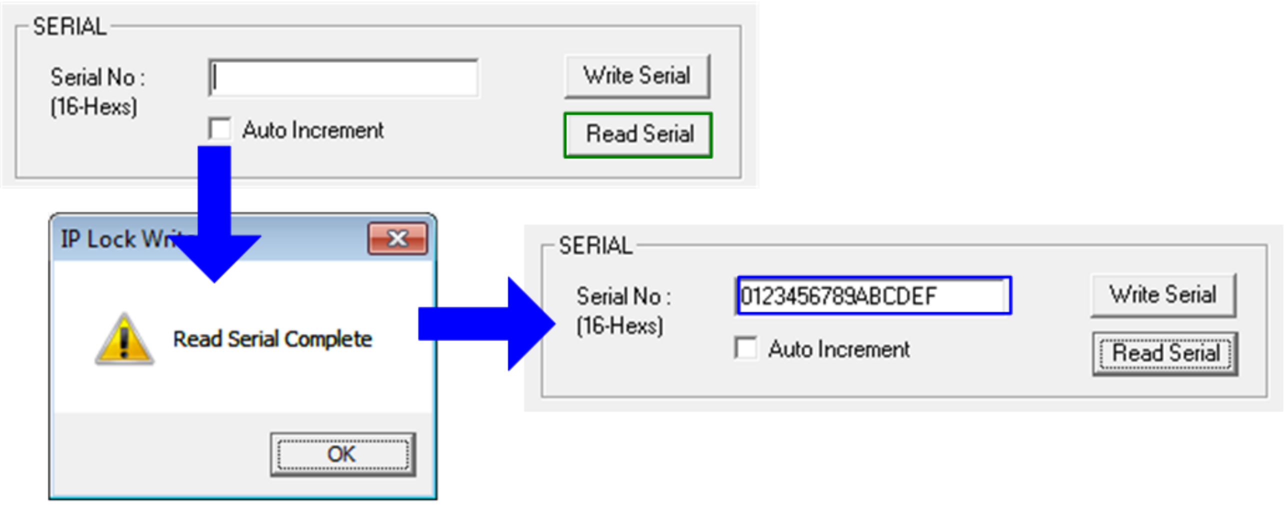


Your serial's length is not full (16-Hexs)
Do you want to use serial : 0123456789ABCDEF

Write Serial complete

Figure 4-13 Manual Assignment mode

Second is Manual assignment mode ("Auto Increment" box is not checked). The user can fill 64 bit Serial number as hexadecimal digit. If the user inputs are less than 16 hex digits, the software will fill 0x00 in the upper bit. As shown in Figure 4-13, when the input is 0x123456789ABCDEF, the software will update the input to 0x0123456789ABCDEF. Similar to Auto increment mode, "Write Serial complete" dialog is displayed after finishing the operation. In this mode, the value in Serial No box does not change after the operation completes.



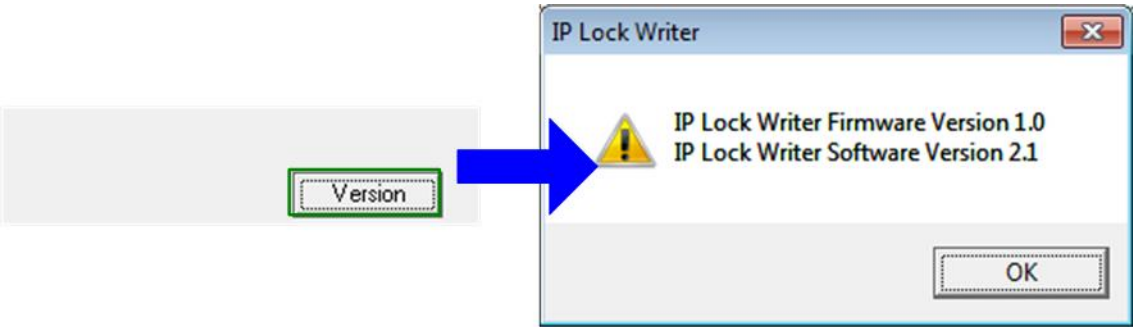
Read Serial Complete

Serial No : 0123456789ABCDEF

Figure 4-14 Read Serial number

To read Serial number, the user clicks “Read Serial” button. “Read Serial Complete” dialog is displayed when finishing the operation. As a result, the read value from IP Lock device is shown on the box. The factory default value of Serial number is 0xFFFFFFFFFFFFFFFF.

4.3.3 VERSION



IP Lock Writer Firmware Version 1.0
IP Lock Writer Software Version 2.1

Figure 4-15 Read Version

The user can click “Version” button to check the firmware version and the software version of IP Lock Writer, as shown in Figure 4-15.

5. Troubleshooting

The following information is the conclusion of some problems which may be found when using the IP Lock product.

Q: Why the ENABLE signal output from IP Lock core is de-asserted to '0'?

A: (1) Please check IP Lock device direction on PCB is mounted correctly.

(2) Please check pin assignment of the FPGA is correct position.

(3) Please check the user's key in IP Lock device and IP Lock core are same value.

Q: What is the specification of power supply voltage for IP Lock device?

A: Please supply voltage of IP Lock device by using the same source with FPGA I/O pins which are connected to IP Lock device.

Q: Do DD0 and DC0 of IP Lock device need pull up?

A: The user can add the pull up registers on PCB or use the internal pull up by FPGA assignment.

Q: Why the IP Lock core and the IP Lock writer must use from the same product box?

A: Because the IP Lock core and the IP Lock writer have the writer's ID which is the unique value for each product box. As a result, though user writes same user's ID to the IP Lock core and the IP Lock writer, the failure still be found from the mismatch writer's ID.

If the above suggestion cannot solve your problem, please contact our support team by using email or sending the inquiry on our website.

IP Lock product support email: iplock@design-gateway.com

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Security/Authentication Development Tools ***category:***

Click to view products by Design Gateway ***manufacturer:***

Other Similar products are found below :

MAX16984EVKIT# MAXREFDES132# DS28E25EVKIT# MAX20010EVKIT# MAX98091EVKIT#TQFN IPL-003WR ATCRYPTOAUTH-XPRO BLOCKCHAINSTARTKITTOBO1 IRID9670TPM12LINUXTOBO1 OPTIGATRUSTEEVALKITTOBO1 OPTIGATRUSTMEVALKITTOBO1 MAX66240EVKIT# MAXAUTHDEMO1# DS28C16EVKIT# DS28C39EVKIT# DS28C40EVKIT# DS28C50EVKIT# DS28E15EVKIT# DS28E16EVKIT# DS28E22EVKIT# DS28E39EVKIT# DS28E50EVKIT# DS28E83EVKIT# DS28E84EVKIT# DS28EL15EVKIT# MAXREFDES143# MAXREFDES43# MAXREFDES44# AT88CK590 AT88CKECC-AWS-XSTK-B ATCRYPTOAUTH-XPRO-B DM320109 DM320118 DT100104 MIKROE-3746 MIKROE-3774 MIKROE-3915 MIKROE-4236 MIKROE-1819 MIKROE-2829 MIKROE-3045 OM3710/A71CHARD OM-SE050ARD 102010288 AT97SC3205P-SDK2 ATECC108XPLAINED MAXREFDES34# DS28C36EVKIT# DS28E38EVKIT# AT88CKECC-AWS-XSTK