

Rockwell Software Studio 5000 ControlLogix Programming

Table of Contents

	Page
A. Starting a Project With the ControlLogix	Studio5000–3
1. Configuring the Processor	Studio5000–3
2. I/O Configuration	Studio5000–3
3. Specifying Other Tags	Studio5000–5
4. Processor Channel Configuration	Studio5000–5
5. Processor Download	Studio5000–5
B. Ladder Logic Programming	Studio5000–6
1. Two Contacts in Series (Logical AND)	Studio5000–6
2. Entering Branches (Logical OR)	Studio5000–7
3. Timers	Studio5000–8
4. Counters	Studio5000–8
5. Online Editing With PLC in Run Mode	Studio5000–8
C. Adding Comments to Ladder Logic	Studio5000–10
D. Displaying Memory Locations	Studio5000–11
E. Input/Output Forcing	Studio5000–11
F. Importing	Studio5000–12
1. Importing Database from CSV files	Studio5000–12
2. Importing PLC Program	Studio5000–12
G. Saving Ladder Logic Program	Studio5000–13
H. Restoring Ladder Logic Program	Studio5000–13
I. Printing Ladder Logic Program	Studio5000–14

This document is a supplement to the “Lab 1.doc” lab exercise that introduces one to the basics of ladder logic programming. The following equipment is assumed:

A-B 1756-A10 10-slot chassis

A-B 1756-P___ power supply

A-B 1756-L71 (Rev 34) processor in slot 0

A-B 1756-IB32 (Rev A) discrete input module in slot 1

A-B 1756-OB16I (Rev A) discrete output module in slot 2

This document is primarily for Rev 34 of the Studio 5000 software. A USB cable connection from the PC to the processor is also assumed.

In order to become familiar with basic ladder logic programming and addressing techniques on the ControlLogix processors, the steps to accomplish the first part of the first lab exercise are outlined in sections A and B. The remainder of the sections outline the steps to document your ladder, to print the ladder logic program, and to save your ladder logic diagram.

In an industrial setting, the initial PLC programming is done offline and then later downloaded to the PLC when ready for the final test and debug. In addition, there is usually only one project (program) for each PLC on the network. For these labs, the ladder programming will be done online so that changes can be immediately tested.

Rockwell Software Studio 5000 is used to program the Allen-Bradley ControlLogix. An icon for this program should already be on the desktop.

The PLC and the PC used to program it are connected directly through a serial RS-232 link.

For the casual user, the RSLogix software is best operated using the point-and-click approach. Towards the top of the screen, buttons for frequently-used commands are displayed in toolbars. One may click on the button, or access the function by selecting commands from the menu at the top of the screen. The various functions of the software may also be accessed by typing various letter mnemonic commands, but unless you use the software extensively, the learning curve is too steep for this laboratory.

A. STARTING A PROJECT WITH THE CONTROLLOGIX

Wire the switch/light board before proceeding. Power up the computer, the PLC rack, and external power supply, as needed. Turn the key on the PLC to the REM (middle) position.

If the Studio 5000 Programming Software main menu is not already displayed, click on the Studio 5000 icon. The main menu screen should appear with a list of selections.

For a new project, click on "New Project" beneath "Create."

If your project already exists, select it from the "Recent Projects", if it appears in the list, or click on "Existing Project", select "Project File" and browse to your project (*.acd) file.

1. Configuring the Processor

If no processors appear in the right side of the pop-up window, click on "Logix" on the left side.

If the "ControlLogix® 5570" line does not show processors below it, click on the arrow at the front of the line. Select the proper processor type:

1756-L71 ControlLogix® 5570 Controller

Type in a controller name.

Browse to the proper location folder in which the project will be saved (Should be on S:\).

Click on *Next*.

Select the proper revision level, **34**.

Select the proper chassis type (7- or 10-slot)

Select slot for the processor (probably "0").

Click on *Finish*.

The project data bases will be created. A controller organizer will appear on the left and a large blank area will appear.

Set up the communications path:

On the main menu bar, select **Communications / Who Active**.

Click on "+" box next to "USB" since the processor is connected with a USB cable to the PC.

Select the line with "1756-L71 LOGIX5571".

Wait a few seconds, to allow the "Set Project Path" button to be visible.

Check box **Set Project Path**.

Click on **Close**.

You are now ready to configure the I/O channels. The OFFLINE/ONLINE window (on the left side, towards the top, above the Controller Organizer) displays "Offline".

2. I/O Configuration

Using the Logix Designer software, configuration of the inputs and outputs involves two steps:

1. specifying modules in the rack

2. assigning tags to the I/O channels

The second step is not absolutely necessary, but it is easier to specify tags now than when programming the ladder logic. The programming software aids the programming of a PLC because one does not need to memorize the I/O addresses and file addresses, but can refer to them symbolically, as a tag.

The following directions assume that the following modules are in the chassis:

Slot 1 - 1756-IB32

Slot 2 - 1756-OB16I

Right-click on the "1756 Backplane" line below the I/O Configuration folder in the project tree. A pop-up window will appear. Select **New Module...** A window titled "Select Module Type" will appear. To add the 1756-IB32, the discrete input module,

In the main part of the window, a list of available modules is displayed. One can scroll through the list, but the shortcut is to type "IB32" in the search box towards the top of the popup. Click on the type of module (1756-IB32) to select it.

Click on **Create**.

A window titled "New Module" will appear. Enter a name for the module (Suggestion: discrete_in). Verify that the slot number is correct (slot 1).

Change the revision to "2" as follows:

Click on the **Change...** button

Set the "Series" to "A" and the Revision should be "2"

Check that the "Connection" and "Input Data" are both "Data". If not change them

Click on **OK**.

Click on **Yes**.

Click on **OK** to accept the module configuration.

Slot 1 is now configured for a 32 point discrete input module.

The "Select Module Type" window should still be displayed. If not, right-click on the "1756 Backplane" line below the I/O Configuration folder in the project tree.

Add a 1756-OB16I. The steps are similar to the steps to add the discrete input module. Make sure the module name is different from the discrete input module. The slot number should be "2". The Revision should be "1.001." The Connection and Output Data should both be "Data." If not, click on the **Change...** button to correct the revision.

The "Select Module Type" window should still be displayed. Click on **Close** to close it.

The I/O Configuration folder should have the following subitems:

[1] 1756-IB32/A <name>

[2] 1756-OB16I <name>

Now assign tags to the input channels. Double-click on the **Controller Tags** line in the project tree window (should be second line). The Controller Tags window will appear. To assign symbols to the discrete input channels,

Click on the **Edit Tags** tab (at the bottom).

A table will appear with a white last row. New tags are entered in the last (white) row.

Type your tag name, in this case "Start_PB" in the **Tag Name** column and press <Tab>.

Click on the down arrow in the **Alias For** column.

Click on "Local:1:I" and click on the "►" to display the members of this structure.

Click on "Local:1:I.Data" and then on the down arrow.

Click on the proper bit ("0" in this case).

Note: one could simply type "Local:1:I.Data.0" in the alias column rather than use the mouse.

Press <Tab> three times, or click on the entry for the **Description**. The **Type** for "Start_PB" should be BOOL. Type in a description in the box, for example, "Start push button switch" and then press <Enter> twice or click back on the box in the Description column.

Repeat the above 6 steps to specify the tags and descriptions for the other input and output channels.

3. Specifying Other Tags

It is good practice to enter all of the tags before you start programming. Though for this exercise, you do not yet know what timer/counter tags you will want to create for later parts of this exercise. Fortunately, you can create new tags as you program the ladder logic.

Now the processor needs to be downloaded with the blank ladder.

4. Processor Download

Click on the down arrow in the OFFLINE/ONLINE window. Select **Download**. A Download window will pop up. Click on **Download**. If a window pops up informing you that the communication configuration is different, click on ***Don't Apply***. If a Save Project window appears, click on ***OK***.

If a window pops up, asking to change controller to Remote Run, select ***No***. If a window pops up and asks if you want to go Online, click on ***Yes***.

B. LADDER LOGIC PROGRAMMING


The OFFLINE/ONLINE box should show "REMOTE PROG" backlighted in blue. The steps necessary to program the first part of the experiment, the series rung, will be explained in detail. For the other parts of the exercise, only the different steps required to do these parts are explained.


1. Two Contacts in Series (Logical AND)

The first ladder program to be implemented will be a simple series control rung. Two switches, SW3 and SW4, will be placed in series and will be used to control a single light, LA1.

Use the right mouse button to access editing functions whenever possible. Clicking on the right mouse button brings up the editing menu with available options. The menu is different depending on the (box) cursor location. If the box is on the left rail of a rung, **rung** editing functions: accept, cancel, cut, paste, append rung, insert rung, etc. If the box is on an instruction, then **instruction** editing functions are presented if the right mouse button is clicked: append/insert instruction, insert/append branches, change instruction type, cut, paste, etc. As you place the instructions into a ladder program, one can either *append* or *insert* the instruction. Append places the new instruction to the right of the current cursor position, and insert places the instruction to the left. If the cursor is on the left power rail, append or insert will append or insert a rung. *Appended* rungs are placed *below* the current rung, and *inserted* rungs are placed *above* the current rung.

If ladder window is not displayed, double-click on "MainRoutine" under "Main Task" in the project window. Place the cursor (blue box) on the END rung. If not, click on the left rail of the bottom rung.


Click on the Insert Rung icon (). A new rung appears at the top of the screen with a blue box on the left. The "e" in the box indicates this is an edit zone. Now the individual instructions for the new rung will be entered.

A normally open (NO) relay contact is needed at the input, so click on the appropriate contact symbol () above the ladder window.


Move the cursor arrow tip close to the left side of the rung and click the mouse. The NO contact will appear with a box around it and a question mark above it.

Switch SW3 will be used first, so type the tag for that switch ("SW3") and press <Enter>.

The symbolic and physical address of the relay that controls the contact will appear above the contact symbol on the ladder diagram.

Click on the  contact symbol above the ladder window and then click on the new rung to the right of the first contact and add a second NO relay in series with the first. Use the symbol or address for switch SW4 ("SW4") for this contact.

The output for this rung will be light LA1.

Click on the output coil symbol () above the ladder window.

Move the cursor to the right side of the rung and click the mouse. A normal output coil symbol will appear with a box above it.

Type the symbol or address for light LA1 ("LA1").

Since the output instruction has been entered, the rung edits may now be accepted.

Click on the left rail of the rung. A blue box appears.

Click the right mouse button.

Click on ***Accept Rung Edits***. Now the rung is loaded into the controller's memory. The "e" indications on the left side of the rung should disappear.

To change the processor to the run mode use the following keystrokes:

Click on the arrow to the right of "REMOTE PROG" box

Click on ***Run***

Click on ***Yes***

The controller is now set to respond to input conditions and perform appropriate output commands. The ladder power rails change to green, indicating the processor in the run mode.

Try the input switches in various combinations and observe the output. Also observe the changing status of ladder components on the screen. Energized or true conditions cause the elements to be highlighted on the ladder diagram.

2. Entering Branches (Logical OR)

Change the processor back to program mode:

Click on the arrow to the right of "REMOTE RUN" box

Click on ***Program***

Click on ***Yes***

Either insert a new rung, or start rungs edits on an existing rung, placing the cursor on the instruction that will parallel the new branch. To start the branch:

Place the cursor on the contact that will parallel the new branch.

Click on the branch icon (looks like a "branch") above the ladder window or right click the mouse and select ***Insert New Branch***

A branch will appear to the right of the cursor with a blue box surrounding the right side of the branch

Place the mouse cursor on the blue box and hold down the mouse button as you drag the branch to the desired location of this side of the branch, usually on the other side of an instruction. The blue oval on the rung that changes to green indicates the location of this branch leg. Release the mouse button and the program will place the branch.

Instructions are entered on branches just as if they were a separate rung.

To make a series of parallel branches that have parallel sides (instead of nesting, which looks like an inverted tree), the branch is extended down.

Place the cursor on the lower left corner of the branch.

Right click the mouse

Select ***Extend Branch Down***

A branch will appear below the existing branch and parallel with the existing branch

3. Timers

To enter a timer into a rung, position the cursor on the right side of the rung, or to the left of where you want to place the instruction.

Click on the Timer/Counter tab above the ladder window and click on **TON**.

A timer instruction will appear at the right side of the rung. A box with a cursor will appear.

Type the tag in the "Timer" field and press <Enter>

Type the preset value and press <Enter>

Type the initial accumulator value and press <Enter>

Other types of timer instructions are selected by clicking on the Timer/Counter tab and then the appropriate mnemonic (TON, TOF, RTO, or RES (reset timer)).

Note: to refer to the timer done bit in a contact address, use "timer_tag.DN".

4. Counters

Counter blocks are placed in the ladder rung in much the same way as timers. To enter an up counter into a rung, first click on the **CTU** and position the arrow on the right side of the rung.

Click the mouse.

A CTU instruction will appear at the right side of the rung. A box with a cursor will appear.

Type the tag in the "Counter" field and press <Enter>

Type the preset value and press <Enter>

Type the initial accumulator value and press <Enter>

For other types of counter instructions are selected by clicking on the Timer/Counter tab and then the appropriate mnemonic (CTU, CTD, or RES (reset timer)).

Note: to refer to the counter done bit in a contact address, use "counter_tag.DN".

5. Online Editing With PLC in Run Mode

The ladder logic program may be edited while the PLC is in the Run mode. A few more steps are required, since accepted rungs are not immediately executed. The process of online editing is illustrated by Figure 1. After rungs are edited, they must be tested and then assembled before the process is complete. If the edits are discovered to be unacceptable while in the "Test Edit" mode, the edits may be untested and modified, or cancelled.

While online editing a rung (edit, insert, replace, or delete), edit zone markers are placed in the left margin. These indications are explained below. While a rung is being edited, the zone markers are lower case, indicating that the edited rung exists only in the computer memory, not in the PLC. After a rung is accepted, the edit zone markers change to upper case letters, indicating that the edited rung is now in the PLC, but not being executed. Other rungs may be edited. After edited rungs are accepted, the old (R and D edit zones) and new rungs (I edit zones)

co-exist in the PLC. However, only the old rungs are executed. The new rungs are skipped, indicated by the edit zone markers in the left margin.

To execute the new rungs, the edits must be tested (right mouse, **Test Edits**). While the edits are being tested, the new rungs (I edit zones) are executed and the old rungs (R and D edit zones) are skipped.

If the program operation with the new rungs is acceptable, then the new rungs should be assembled (right mouse, **Assemble Edits**). When the rungs are assembled, the old rungs are deleted from the PLC memory and the edit zone markers are removed.

If the new rungs are unacceptable, then the new rungs should be untested (right mouse, **Untest Edits**). When the rungs are untested, rungs are restored to their state before the rungs were tested. The old rungs (R and D zones) are executed and the new rungs (I zones) are skipped.

After the rungs have been untested, or before the edits are tested, the edits may be cancelled (right mouse, **Cancel Edits**). When the edits are cancelled, the new rungs (I zones) are deleted and the edit zone markers are removed.

Lower-case edit zone markers

- e** (Offline and online) These rungs are currently being edited and are contained in the computer memory. If one is working offline, the edit zone marker disappears after the rung is successfully verified (right mouse button selection). If one is working online, the edit zone marker is changed to an upper-case I after the rung is successfully accepted (right mouse button selection).
- i** (Online) These rungs will be inserted into the PLC program, but exist in the computer memory, not in the PLC memory. After the rung is successfully accepted (right mouse button selection) the rung is inserted into the PLC memory and the edit zone marker is changed to an upper-case I.
- r** (Online) These rungs will be replaced in the PLC program. An r marked rung is always preceded by an e marked rung. After the e marked rung is successfully accepted (right mouse button selection) the r edit zone marker is changed to a R edit zone.
- d** (Online) These rungs will be deleted from the PLC program. However, the lower-case d indicates that the deletion only exists in the computer memory. When the rung is accepted (right mouse button selection), the lower-case d is replaced by an upper-case D, indicating that the deletion exists in the PLC memory.

Upper-case edit zone markers

- I** (Online) These rungs are inserted into the PLC program. To test the rung execution, select "Test Edits" (right mouse button selection). Select "Assemble Edits" to finalize rung insertion and to remove the edit zone markers.
- R** (Online) These rungs are replaced in the PLC program. However, an R marked rung is still executed until "Test Edits" is selected. When "Assemble Rungs" is selected, the rung is deleted.

- D** (Online) These rungs are deleted from the PLC program. However, a D marked rung is still executed until "Test Edits" is selected. When "Assemble Rungs" is selected, the rung is deleted.

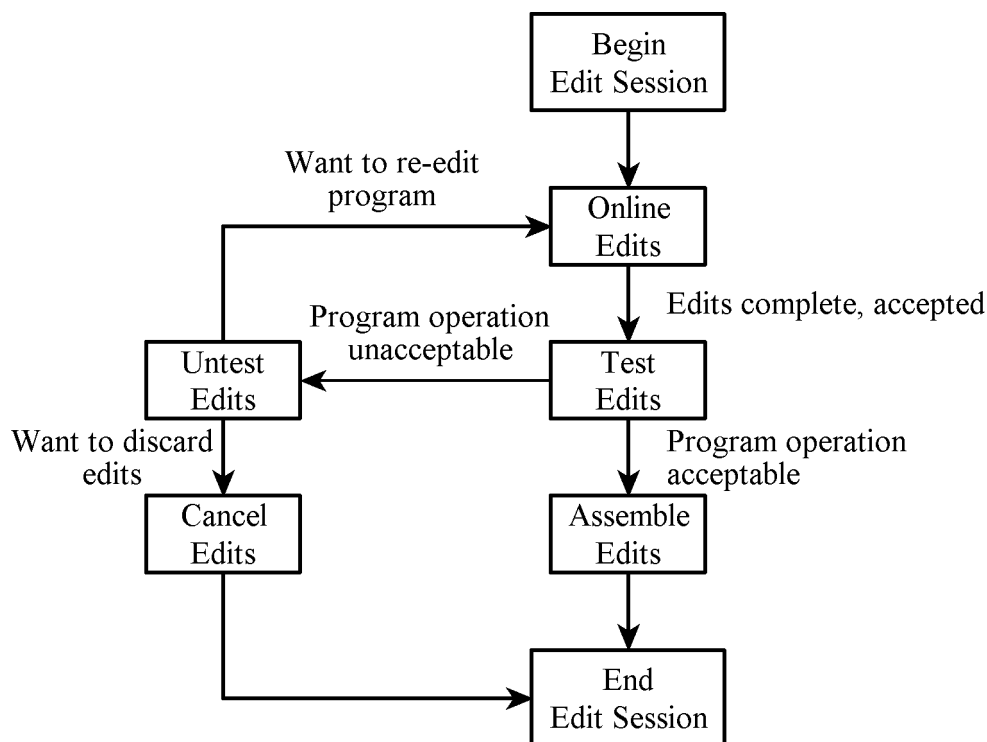


Figure 1. Online editing steps.

C. ADDING COMMENTS TO LADDER PROGRAM

Select the rung you wish to comment by placing the box cursor on the left power rail of the rung, or on any instruction in the rung.

Right-click on the left power rail for the rung

Select ***Edit Rung Comment***

A window opens with a vertical cursor. Type descriptive text. Two consecutive <Enter> keys will terminate the comment.

To change an existing rung comment double click on the comment box and a text window opens with the comment. Make your changes. Click on the green check mark when finished.

To display the rung comments on the ladder logic display, the "Display page titles/rung descriptions" must be enabled. From the main menu:

Select ***T*ools**

Select ***O*ptions...**

Expand the *Ladder Editor* line
 Select *Display*
 Make sure "Show Rung Comments" is check marked

D. DISPLAYING MEMORY LOCATIONS

The *watch* window allows one to display and monitor a few tags without searching through the entire tag database. This feature is very useful for debugging program operation and mathematical operations. To display the watch window, select **View | Watch** from the Studio5000 main menu. The watch window appears below the ladder logic window. If the watch list is set to "Current Routine", all of the tags used in the displayed routine along with their current values are shown in the watch window. If the watch list is set to "Quick Watch", one specifies the tags to be monitored in the "Name" column.

To change the value of a tag in the watch window, double-click in the "Value" column. Now change the value.

To change the data of a tag, first display the tag. If it is a structure, you can only change individual structure elements. Click on "+" to show the elements. Use the mouse to put the cursor into the (white) value box, and type the new number, followed by <Enter>.

One can also display the tag values by browsing the Controller Tags or Program Tags. Double-click on the Controller Tags or Program Tags (depending on location of tag) to open the data monitor window. Click on the Monitor Tags tab to display the values of the tags.

E. INPUT/OUTPUT FORCING

For inputs, the force function allows one to override the actual status of external inputs by forcing individual discrete inputs to be ON or OFF. Input forces are written to the input image table and have no effect on the actual input. The value in the force table overrides the actual input. For example, if Local:0:I.Data.13 is forced to 1 (ON), it will appear in the input image and ladder logic as 1, regardless of the actual status of the device.

For outputs, the force function allows one to override the processor logic and force individual discrete outputs to be ON or OFF. Output forces are written directly to the output device and do not appear in the output image table. The value in the force table overrides the value in the output image. For example, if Local:4:O.Data.2 is forced to 1 (ON), the actual output will be ON, regardless of how it appears in the ladder and output image table.

Note: Enabling or disabling forces can result in sudden machine movement, possibly injuring personnel. Applying or removing forces while forces are enabled takes place immediately, with no prompt for confirmation. **USE EXTREME CAUTION WHEN USING FORCES!**

One can program forces in one of two ways:

- Position the cursor on an instruction in the ladder program which references a real I/O address and from the right-mouse menu select ***Force On*** or ***Force Off***.
- Edit the input or output force from the data monitor window. The force is in the “Force Mask” column. Set the force using either the tag name aliased to the physical I/O, or access the individual bit in the I/O module data tag. The current state of the force is indicated as:

(blank)	there is no force for this address
1	address is forced ON
0	address is forced OFF

To change the force status of the bit, edit the entry in the “Force Mask” column.

To enable forces, locate the forces list box just below the mode selection box. Select ***Forces / I/O Forcing / Enable All Forces***. A confirmation window will appear. Click on ***Yes***.

To temporarily disable all forces, select ***Forces / I/O Forcing / Disable All Forces*** from the force list box. A confirmation window will appear. Click on ***Yes***.

To remove all forces, select ***Forces / I/O Forcing / Remove All Forces*** from the force list box. You will be asked if you want to remove all forces. Click ***Yes*** to remove all forces.

F. IMPORTING

1. Importing Database from CSV files

To import an address/tag/description database from a .CSV (comma separated variable) file:

From the main menu select ***Tools***.

Select ***Import Tags...***

An Import Tags window will appear. Select the following:

Files of Type - ***CSV***

Select the appropriate file.

On Collisions: - ***Overwrite***

Create tag if it doesn't exist - checked

Click on ***Import***

The addresses and tags are imported.

A window showing the status of the import will be displayed. Click on ***OK*** when finished.

2. Importing PLC Program

To import ladder logic rungs contained in a library (.L5X) file: (This procedure also imports tags used in the ladder logic)

Place your cursor where you want the ladder logic to be added. It will be placed before the cursor. If you want to add it to the end of the program, place the cursor on the (End) rung.

Right-mouse the left side and select ***Import Rung..***

Select the appropriate L5X file and click ***Import***

An import configuration window appears. This is your opportunity to change tags, but normally not necessary. Check the extreme left side. Any tags already defined in the project will be indicated with a yellow triangle.

Click on ***OK*** Any tag collisions are also noted in the "Errors" window at the bottom of the screen.

G. SAVING LADDER LOGIC PROGRAMS

Ladder logic programs (projects) may be saved for future use. By saving your ladder logic programs, you are able to resume your experiment, even though others have been using the same PLC. One usually saves the program to the default program directory on the hard disk and then saves the program to a USB flash drive or your personal account.

To save the ladder logic currently in the PLC,

Select ***File***

To save with the same project name, select ***Save***

To save with a different project name, select ***Save As...***

To save your project to another directory, use ***Save As...*** and select the proper drive and directory.

H. RESTORING LADDER LOGIC PROGRAMS

You may restore your ladder logic program from the files on the computer hard disk to the PLC either before going on-line. Before going on-line, do the following steps:

Select ***File***

Select ***Open***

The file selector window will appear.

Double-click on the appropriate project. The project is now loaded into the PC

From the "Offline" box in the upper left part of the screen, select ***Download***

A window will pop up asking for confirmation. Select ***Yes***

If the processor is in the run mode, you will be asked if it is okay to switch to program mode before continuing. Select ***Yes***

The files will be downloaded

If the processor was in the run mode when you started, you will be asked if it is okay to switch to back to run mode before continuing. Select ***Yes***

I. PRINTING A LADDER LOGIC PROGRAM

You may print your program only while off-line. To print,

- Right-mouse first line in project tree

- Select ***Generate Report...***

- De-select Tags

- Click on ***Print***