FANUC Communication Driver

Driver for serial communication with
GE FANUC device using SNP protocol

Contents

INTRODUCTION **2**

GENERAL INFORMATION **3**

DEVICE SPECIFICATIONS 3
NETWORK SPECIFICATIONS 3
DRIVER CHARACTERISTICS 4
CONFORMANCE TESTING 5

SELECTING THE DRIVER **6**

CONFIGURING THE DEVICE **7**

CONFIGURING THE DRIVER **7**

CONFIGURING THE COMMUNICATION SETTINGS 7
CONFIGURING THE DRIVER WORKSHEETS 9

EXECUTING THE DRIVER **15**

TROUBLESHOOTING **16**

SAMPLE APPLICATION **18**

REVISION HISTORY **19**

USING LOGICMASTER 90 SOFTWARE TO CONFIGURE COMMUNICATION PARAMETERS **21**

Introduction

The FANUC driver enables communication between the Studio system and GE FANUC devices using the SNP protocol, according to the specifications discussed in this document.

This document will help you to select, configure and execute the FANUC driver, and it is organized as follows:

- **Introduction**: This section, which provides an overview of the document.
- **General Information**: Identifies all of the hardware and software components required to implement communication between the Studio system and the target device.
- **Selecting the Driver**: Explains how to select the FANUC driver in the Studio system.
- **Configuring the Device**: Describes how the target device must be configured to receive communication from the FANUC driver.
- **Configuring the Driver**: Explains how to configure the FANUC driver in the Studio system, including how to associate database tags with device registers.
- **Executing the Driver**: Explains how to execute the FANUC driver during application runtime.
- **Troubleshooting**: Lists the most common errors for this driver, their probable causes, and basic procedures to resolve them.
- **Sample Application**: Explains how to use a sample application to test the FANUC driver configuration
- **Revision History**: Provides a log of all changes made to the driver and this documentation.
- **Using LOGICMASTER 90 Software to configure communication parameters**: Explains how to use LOGICMASTER to configure some communication parameters through a serial connection.

Notes:

- This document assumes that you have read the "Development Environment" chapter in Studio's *Technical Reference Manual*.
- This document also assumes that you are familiar with the Microsoft Windows NT/2000/XP environment. If you are not familiar with Windows, then we suggest using the **Help** feature (available from the Windows desktop **Start** menu) as you work through this guide.

General Information

This chapter identifies all of the hardware and software components required to implement serial communication between the FANUC driver in Studio and a GE FANUC device using the SNP protocol.

The information is organized into the following sections:

- Device Specifications
- Network Specifications
- Driver Characteristics
- Conformance Testing

Device Specifications

To establish communication, your target device must meet the following specifications:

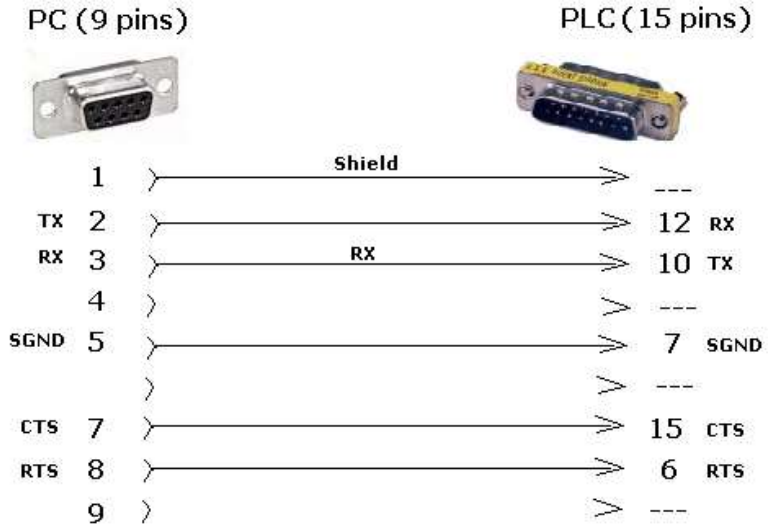
- **Manufacturer:** GE FANUC
- **Compatible Equipment:** PLCs Series 90 – 20/30/70
- **Programmer Software:** LogicMaster 90 Programmer Controller Software, Proficy

For a description of the device(s) used to test driver conformance, see “Conformance Testing”.

Network Specifications

To establish communication, your device network must meet the following specifications:

- **Device Communication Port:** SNP port / CMM311 Serial Comm module - Port1 (IC693CMM311)
- **Physical Protocol:** RS232/RS485 or RS422
- **Logic Protocol:** SNP
- **Device Runtime Software:** None
- **Specific PC Board:** None
- **Link cable scheme:**



- Complete Pin-out (Series 90 PLC):

Pin Number	Signal Name	Description
1	Shield	
2		No Connection
3		No Connection
4	ATCH *	Hand-Held Programmer attach signal
5	+5V *	+5V Power for: HHP and RS-232/485 Converter
6	RTS (A)	Request To Send
7	Signal Ground	Signal Ground, OV
8	CTS (B')	Clear To Send
9	RT *	Terminating Resistor for RD **
10	RD (A')	Receive Data
11	RD (B')	Receive Data
12	SD (A)	Send Data
13	SD (B)	Send Data
14	RTS (B)	Request To Send
15	CTS (A')	Clear To Send

➪ Tip:
Please refer to “Conformance Testing” to see the equipment used in the standard conformance tests for this driver.

Driver Characteristics

The FANUC driver package consists of the following files, which are automatically installed in the **`/DRV`** subdirectory of Studio:

- **`FANUC.INI:`** Internal driver file. *You must not modify this file*.
- **`FANUC.MSG:`** Internal driver file containing error messages for each error code. *You must not modify this file*.
- **`FANUC.PDF:`** This document, which provides detailed information about the FANUC driver.
- **`FANUC.DLL:`** Compiled driver.

Note:
You must use Adobe Acrobat® Reader™ to view the **`FANUC.PDF`** document. You can install Acrobat Reader from the Studio installation CD, or you can download it from Adobe’s Web site.

You can use the FANUC driver on the following operating systems:

- Windows XP/Vista/7
- Windows CE

For a description of the operating systems used to test driver conformance, see “Conformance Testing” below.

The FANUC driver supports the following registers:

Register Type	Length	Write	Read	Bit	Integer	Float	DWord
%I (*Discrete Input*)	1 Bit	•	•	•	–	–	–
%Q (*Discrete Output*)	1 Bit	•	•	•	–	–	–
%M (*Discrete Internal*)	1 Bit	•	•	•	–	–	–
%T (*Discrete Temporary*)	1 Bit	•	•	•	–	–	–
%AI (*Analog Input*)	2 Bytes	•	•	•	•	•	•
%AQ (*Analog Output*)	2 Bytes	•	•	•	•	•	•
%R (*Internal Registers*)	2 Bytes	•	•	•	•	•	•
%G (*Global Genius Data*)	1 Bit	•	•	•	–	–	–

➲ **Attention:**

- Writing bit values to the **`%AI`**, **`%AQ`** and **`%R`** registers is allowed, however it is not contained in the protocol. Before writing, a reading is done, if in this period between reading and the really writing an eventually modification happens in the device, the value will be overlap. Also, when using the Write Trigger, if there are bit addresses on the sheets with these registers, performance may degrade since several read and write operations are necessary, once they are then performed per item.
- Although the protocol supports writing to the **`%I`** and **`%AI`**, beware that these are the physical inputs, so the PLC scan may overwrite these values.

Conformance Testing

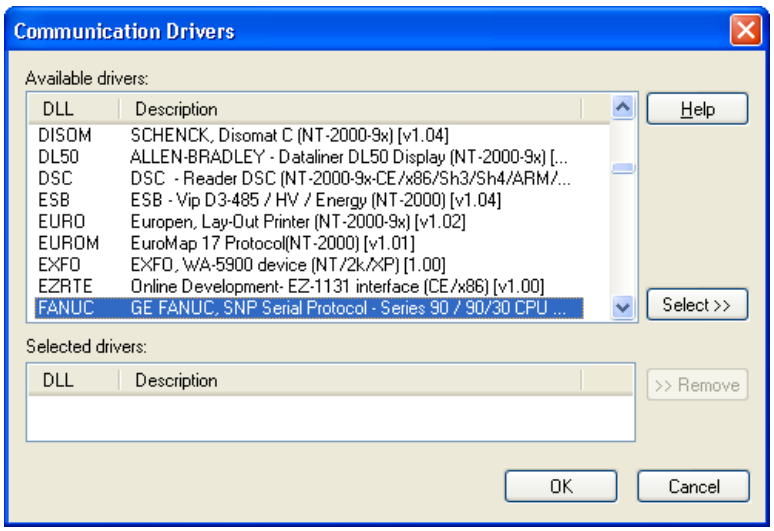
The following hardware/software was used for conformance testing:

- **Equipment**: PLC GE-FANUC 90/30 CPU 341
- **Cable**: Use specifications described in the “Network Specifications” section
- **Operating System** (*development*): Windows XP with Service Pack 2
- **Operating System** (*target*):
  - Windows XP with Service Pack 3
  - Windows CE v5.0
- **Studio Version**: 7.0
- **Driver Version**: 10.4

Selecting the Driver

When you install Studio, all of the communication drivers are automatically installed in the **\DRV** subdirectory but they remain dormant until manually selected for specific applications. To select the FANUC driver for your Studio application:

1. From the main menu bar, select **Insert** → **Driver** to open the *Communication Drivers* dialog.
2. Select the **FANUC** driver from the *Available Drivers* list, and then click the **Select** button.



Communication Drivers Dialog

3. When the **FANUC** driver is displayed in the **Selected Drivers** list, click the **OK** button to close the dialog. The driver is added to the *Drivers* folder, in the *Comm* tab of the Workspace.

> **Note:**
> It is not necessary to install any other software on your computer to enable communication between Studio and your target device. However, this communication can only be used by the Studio application; it cannot be used to download control logic to the device. To download control logic to a GE FANUC device, you must also install the GE FANUC programming software (e.g., LogicMaster). For more information, please consult the documentation provided by the device manufacturer.

> **Attention:**
> For safety reasons, you must take special precautions when installing any physical hardware. Please consult the manufacturer's documentation for specific instructions.

Configuring the Device

Use the following default configuration:

- **Baud Rate:** 19200
- **Data Bits:** 8
- **Stop Bits:** 1
- **Parity:** Odd

You are also required to set the PLC station value, using the programming tools (e.g. Proficy)

Configuring the Driver

Once you have selected the FANUC driver in Studio, you must properly configure it to communicate with your target device. First, you must set the driver's communication settings to match the parameters set on the device. Then, you must build driver worksheets to associate database tags in your Studio application with the appropriate addresses (registers) on the device.

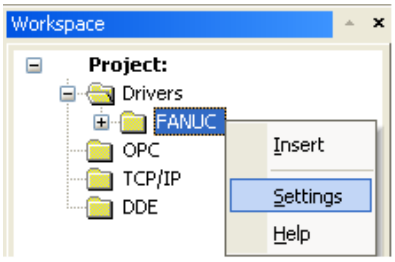
Configuring the Communication Settings

The communication settings are described in detail in the "Communication" chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

For the purposes of this document, only FANUC driver-specific settings and procedures will be discussed here. To configure the communication settings for the FANUC driver:

1. In the *Workspace* pane, select the *Comm* tab and then expand the *Drivers* folder. The FANUC driver is listed here as a subfolder.

2. Right-click on the *FANUC* subfolder and then select the **Settings** option from the pop-up menu. The *FANUC: Communication Parameters* dialog is displayed:



Select Settings from the Pop-Up Menu

FANUC: Communication Parameters Dialog

3. In the *Communication Settings* dialog, configure the driver settings to enable communication with your target device. To ensure error-free communication, the driver settings must *exactly match* the corresponding settings on the device. Please consult the manufacturer's documentation for instructions how to configure the device and for complete descriptions of the settings.

   Depending on your circumstances, you may need to configure the driver *before* you have configured your target device. If this is the case, then take note of the driver settings and have them ready when you later configure the device.

> **Attention:**
> For safety reasons, you **must** take special precautions when connecting and configuring new equipment. Please consult the manufacturer's documentation for specific instructions.

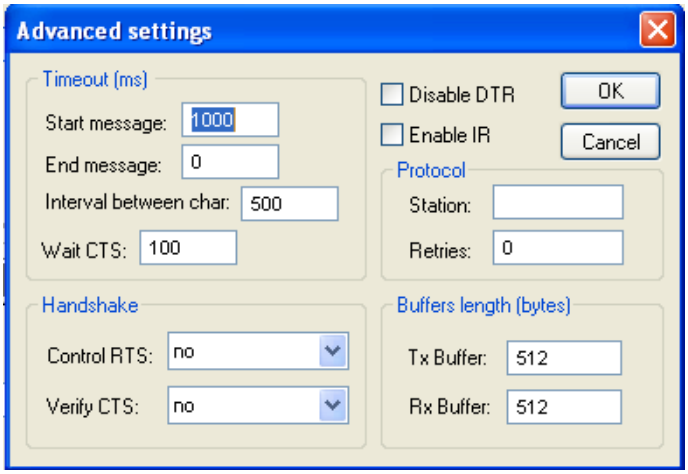
The communication settings and their possible values are described in the following table:

Parameter	Default Value	Valid Values	Description
Yield (0-Yes, 1-No)	0	0 or 1	Internal control. Keep this parameter with the default value: **0**.
0-Signed > 1-Unsigned Value	0	0 or 1	Set the values as signed (**0**) or unsigned (**1**)
Attach Time	1000	From 0 up to 999999	The message of "attach" will be sent if there is no communication in xxxxxx ms. If *Attach Time* is configured with **0** (zero), every communication (read and write) will be sent an "attach," decreasing performance of communication.
Block Size	128	From 128 up to 512	Maximum number of operands per Driver Sheet For values greater than **128**, change to the size of *TX Buffer* and *Rx Buffer* in *Communication Parameters* (*Advanced* to twice the *BlockSize*).
T1	25	From 1 up to 999999	**T1** is the amount of time that must elapse between the reception of the last character or transmission of one message, and the transmission of the first character of the next message. It is the minimum amount of time the sending device (master or slave) must wait before transmitting either a message or an Ack.

> **Attention**:
> Specifying a value for the **Attach Time** parameter that is lower than the default value of `1000` can decrease performance. We recommend using the default value.

> **Note:**
> The device must be configured with *exactly the same* parameters that you configured in the *FANUC Communication Parameters* dialog.

4. If you are using a Data Communication Equipment (DCE) converter (e.g., 232/485) between your PC and your target device, then you must also adjust the **Control RTS** (Request to Send) setting to account for the converter. In the *Communication Settings* dialog, click the **Advanced** button to open the *Advanced Settings* dialog:



Advanced Settings Dialog

When the dialog is displayed, configure the **Control RTS** setting using the following information:

Setting	Default	Values	Description
Control RTS	no	no	Do not set the RTS (Request to Send) handshake signal. IMPORTANT: If you are using Windows 95/98 or Windows CE with the correct RS232/RS485 adapter (i.e. without RTS control), then you must select this option.
		yes	Set the RTS (Request to Send) handshake signal before communication. IMPORTANT: If you are using Windows NT and the Cutler-Hammer RS232/RS485 adapter, then you must select this option.
		yes+echo	Set the RTS (Request to Send) handshake signal before communication, and echo the signal received from the target device.

➲ **Attention**:
If you incorrectly configure the **Control RTS** setting, then runtime communication will fail and the driver will generate a –15 error. See "Troubleshooting" for more information.

You do not need to change any other advanced settings at this time. You can consult the Studio *Technical Reference Manual* later for more information about configuring these settings.

5. Click **OK** to close the *Advanced Settings* dialog, and then click **OK** to close the *Communication Parameters* dialog.

Configuring the Driver Worksheets

Each selected driver includes a Main Driver Sheet and one or more Standard Driver Worksheets. The Main Driver Sheet is used to define tag/register associations and driver parameters that are in effect at all times, regardless of application behavior. In contrast, Standard Driver Worksheets can be inserted to define additional tag/register associations that are triggered by specific application behaviors.

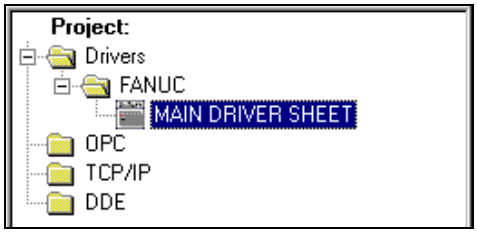
The configuration of these worksheets is described in detail in the "Communication" chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

For the purposes of this document, only FANUC driver-specific parameters and procedures are discussed here.

MAIN DRIVER SHEET

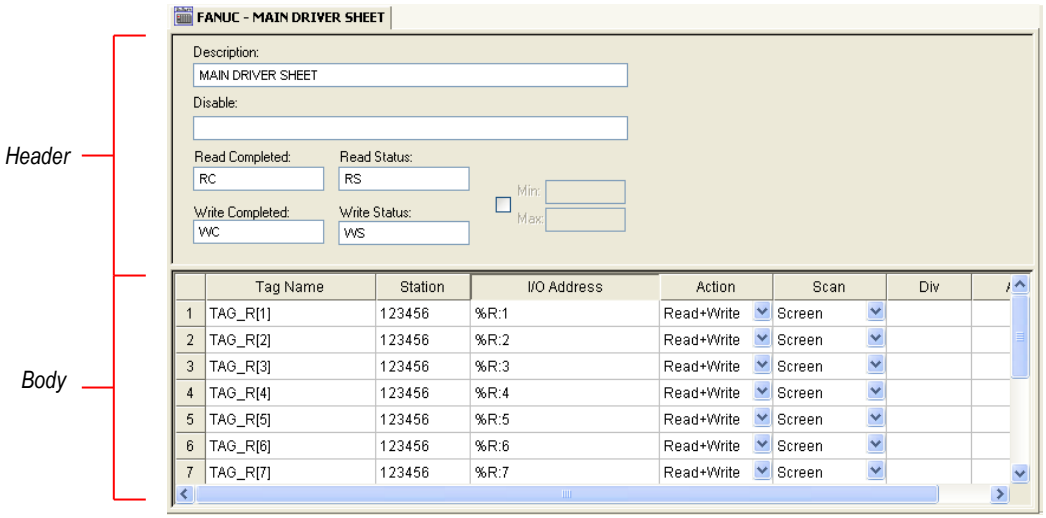
When you select the FANUC driver and add it to your application, Studio automatically inserts the *Main Driver Sheet* in the *FANUC* driver subfolder. To configure the Main Driver Sheet:

1. Select the *Comm* tab in the *Workspace* pane.
2. Open the *Drivers* folder, and then open the *FANUC* subfolder:



Main Driver Sheet in the FANUC Subfolder

3. Double-click on the **MAIN DRIVER SHEET** icon to open the following worksheet:



Opening the Main Driver Sheet

Most of the fields on this sheet are standard for all drivers; see the "Communication" chapter of the *Technical Reference Manual* for more information on configuring these fields. However, the **Station** and **I/O Address** fields use syntax that is specific to the FANUC driver.

4. For each table row (i.e., each tag/register association), configure the **Station** and **I/O Address** fields as follows:

   - **Station** field: Identify the target device, using the following syntax:

     <Device ID>

Example — **938823**

Where ***<Device ID>*** is the name or number of the target device on the serial network. This ID can be any string up to six characters, as determined by the device configuration.

You can also specify an indirect tag (e.g. **{station}**), but the tag that is referenced must follow the same syntax and contain a valid value.

> ➲ **Attention:**
> You must match exactly the PLC configuration in the **Station** field, even if it means leaving the field blank.

- **I/O Address:** Specify the address of the associated device register.

For *Discrete Input*, *Discrete Output*, *Discrete Internal*, *Discrete Temporary* and *Global Genius Data* types, use the following syntax:

<Register Type>*:*<Address>

Example — **%M:100**

For *Analog Input*, *Analog Output* and *Internal Register* types, use the following syntax:

<Register Type>*:*[Data Type]<Address>*.*[Bit]

Example — **%R:10.2**

Where:

– ***<Register Type>*** : GE FANUC register type. Valid values are **%I** (Discrete Input), **%Q** (Discrete Output), **%M** (Discrete Internal), **%T** (Discrete Temporary), **%AI** (Analog Input), **%AQ** (Analog Output), **%R** (Internal Register), and **%G** (Global Genius Data).

– ***[Data Type]*** (optional): Type of analog data. Valid values are **DW** (Double Word) and **F** (Float). Although this parameter is optional, if no type is specified then it is Word by default.

– ***<Address>*** : Address of the device register.

– ***[Bit]*** (optional): The bit number (from 0 to 15) of the address.

Register Type	Valid Range of Address	Comments
%I	1 to 512	Discrete Input
%Q	1 to 512	Discrete Output
%M	1 to 1024	Discrete Internal
%T	1 to 256	Discrete Temporary
%AI	1 to 1024	Analog Input
%AQ	1 to 256	Analog Output
%R	1 to 9999	Internal Registers
%G	1 to 1280	Global Genius Data (Boolean)

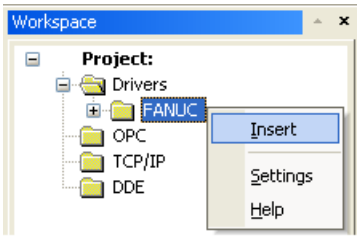
STANDARD DRIVER WORKSHEET

When you select the FANUC driver and add it to your application, it has only a Main Driver Sheet by default (see previous section). However, you may insert additional Standard Driver Worksheets to define tag/register associations that are triggered by specific application behaviors. Doing this will optimize communication and improve system performance by ensuring that tags/registers are scanned only when necessary – that is, only when the application is performing an action that requires reading or writing to those specific tags/registers.

> **Note:**
> We recommend configuring device registers in sequential blocks in order to maximize performance.

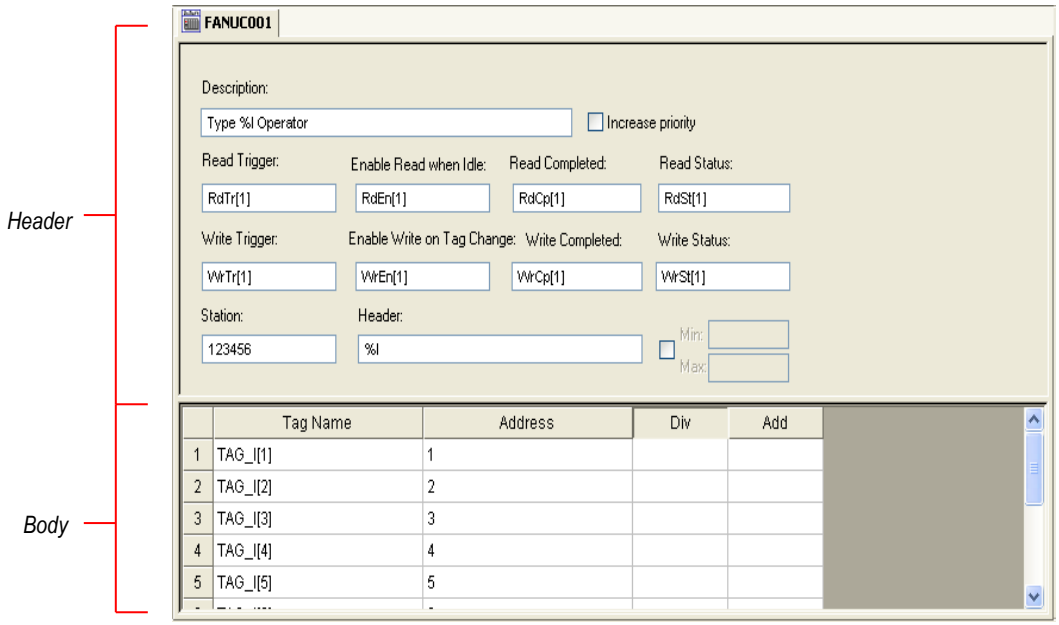
To insert a new Standard Driver Worksheet:

1. In the *Comm* tab, open the *Drivers* folder and locate the *FANUC* subfolder.
2. Right-click on the *FANUC* subfolder, and then select **Insert** from the pop-up menu:



Inserting a New Worksheet

A new FANUC driver worksheet is inserted into the *FANUC* subfolder, and the worksheet is opened for configuration:



FANUC Driver Worksheet

> **Note:**
> Worksheets are numbered in order of creation, so the first worksheet is **`FANUC001.drv`**.

Most of the fields on this worksheet are standard for all drivers; see the "Communication" chapter of the *Technical Reference Manual* for more information on configuring these fields. However, the **Station**, **Header**, and **Address** fields use syntax that is specific to the FANUC driver.

3. Configure the **Station** and **Header** fields as follows:

   - **Station** field: Identify the target device, using the following syntax:

     `<Device ID>`

     Example — **`938823`**

     Where ***`<Device ID>`*** is the name or number of the target device on the serial network. This ID can be any string up to six characters, as determined by the device configuration.

     You can also specify an indirect tag (e.g. **`{station}`**), but the tag that is referenced must follow the same syntax and contain a valid value.

     > **Attention:**
     > You must use a non-zero value in the **Station** field, and you cannot leave the field blank.

   - **Header** field: Specify a single register type on the target device. The addresses declared in the *Body* of the worksheet will automatically be of this type. When Read/Write operations are executed for the entire worksheet (see **Read Trigger** and **Write Trigger** above), it scans the entire block of registers from the first address to the last.

     The **Header** field uses the following syntax:

     `<Register Type>`

     Example — **`%R`**

     Where ***`<Register Type>`*** is the GE FANUC register type. Valid values are **`%I`** (Discrete Input), **`%Q`** (Discrete Output), **`%M`** (Discrete Internal), **`%T`** (Discrete Temporary), **`%AI`** (Analog Input), **`%AQ`** (Analog Output), **`%R`** (Internal Register), and **`%G`** (Global Genius Data).

     After you edit the **Header** field, Studio checks the syntax to determine if it is valid. If the syntax is invalid, then Studio automatically inserts a default value of **`%I`**.

     You can also specify an indirect tag (e.g. **`{header}`**), but the tag that is referenced must follow the same syntax and contain a valid value.

     For each table row (i.e., each tag/register association), configure the **Address** field using the following syntax…

For *Discrete Input*, *Discrete Output*, *Discrete Internal*, *Discrete Temporary* and *Global Genius Data* types, use the following syntax:

`<Address>`

Example — **`100`**

For *Analog Input*, *Analog Output* and *Internal Register* types, use the following syntax:

[Data Type]<Address>.[Bit]

Example — **10.2**

Where:

– ***[Data Type]*** (optional): Type of analog data. Valid values are **DW** (Double Word) and **F** (Float). Although this parameter is optional, if no type is specified then it is Word by default.

– ***<Address>*** : Address of the device register.

– ***[Bit]*** (optional): The bit number (from 0 to 15) of the address.

For examples of how device registers are specified using **Header** and **Address**, see the following table:

Address on the Device	Header Field	Address Field
%I0001	**%I**	**1**
%I0010	**%I**	**10**
%I0512	**%I**	**512**
%Q0512	**%Q**	**512**
%M1020	**%M**	**1020**
%T200	**%T**	**200**
%AI060	**%AI**	**60**
%AI060 (bit 0)	**%AI**	**60.0**
%AQ080	**%AQ**	**80**
%AQ080 (bit 3)	**%AQ**	**80.3**
%R0100	**%R**	**100**
%R0100 (bit 7)	**%R**	**100.7**
%R09900	**%R**	**9900**
%G0500	**%G**	**500**
%G0001	**%G**	**1**
%I0001	**%I**	**1**
%I0010	**%I**	**10**

For more information about device registers and addressing, please consult the manufacturer's documentation.

➲ **Attention:**

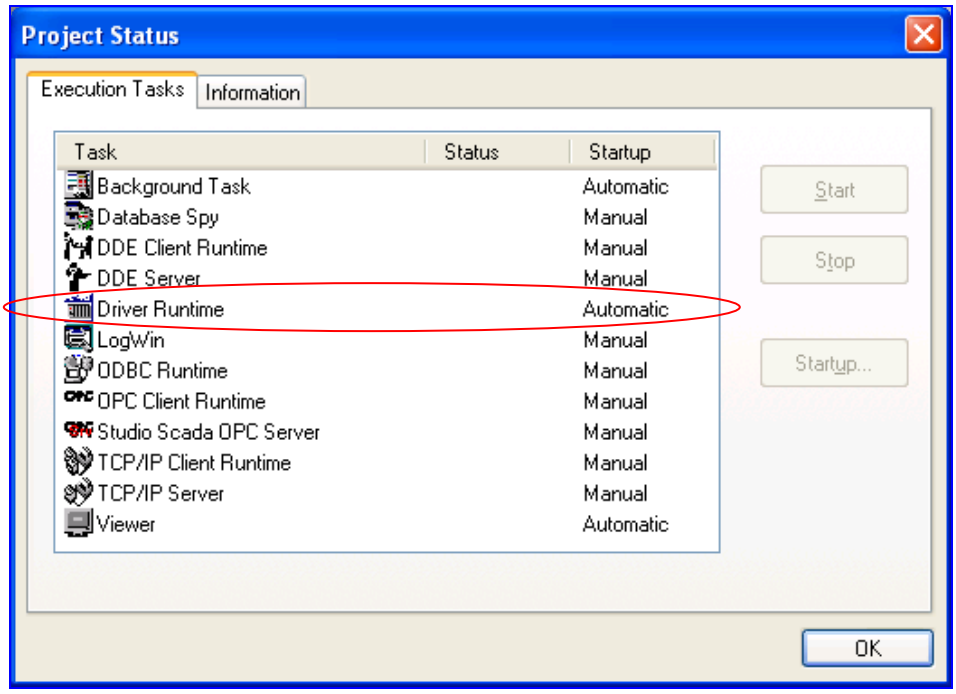
- Writing bit values to the **%AI**, **%AQ** and **%R** registers is allowed, however it is not contained in the protocol. After writing, a reading is done, if in this period between reading and the really writing an eventually modification happens in the device, the value will be overlap.
- You must not configure a range of addresses greater than the maximum block size (data buffer length) supported by each device within the same worksheet. The maximum data buffer length for this driver is 512 bytes per standard driver worksheet.
- The block size for headers **%I**, **%Q**, **%M**, **%T** and **%G** is the number configured on the driver's settings times 8. For example, if the default of 128 is used, the block size for these headers is 1024.

Executing the Driver

By default, Studio will automatically execute your selected communication driver(s) during application runtime. However, you may verify your application's runtime execution settings by checking the *Project Status* dialog.

To verify that the the communication driver(s) will execute correctly:

1. From the main menu bar, select **Project** → **Status**. The *Project Status* dialog displays:



Project Status Dialog

2. Verify that the *Driver Runtime* task is set to **Automatic**.
   - If the setting is correct, then proceed to step 3 below.
   - If the **Driver Runtime** task is set to **Manual**, then select the task and click the **Startup** button to toggle the task's *Startup* mode to **Automatic**.
3. Click **OK** to close the *Project Status* dialog.
4. Start the application to run the driver.

Troubleshooting

If the FANUC driver fails to communicate with the target device, then the database tag(s) that you configured for the **Read Status** or **Write Status** fields of the Main Driver Sheet or Standard Driver Sheet will receive an error code. Use this error code and the following table to identify what kind of failure occurred.

Error Code	Description	Possible Causes	Procedure to Solve
0	OK	▪ Communication without problems	▪ None
1	Invalid Header	▪ Tag placed at the **Header** field with an invalid value	▪ Assign a valid value to this tag
2	Invalid Address	▪ Address with a value less than or equal to zero, or greater than the maximum of this operator	▪ Look at the ranges of the addresses
3	Invalid Block Size	▪ The offset between the minor and major address is greater than the Block Size value	▪ Rearrange the addresses. ▪ Check the Configuration of Block Size. ▪ Increase the Block Size value in Communication Parameters.
4	Invalid Check Sum	▪ Invalid Checksum in response message	▪ Check the cable wiring. ▪ Check the station number. ▪ Check the RTS/CTS configuration (see Studio *Technical Reference Manual* for valid configurations).
5	Nack received	▪ Ack action error during communication	▪ Check device and Studio Communication Parameters.
6	Invalid Frame Number	▪ The driver received the wrong message.	▪ Check device and Studio Communication Parameters.
7	Message was not completed	▪ The driver received the wrong message.	▪ Check device and Studio Communication Parameters.
8	Invalid Check Sum (Attach message)	▪ Invalid Checksum in attach response message	▪ Check the cable wiring. ▪ Check the station number. ▪ Check the RTS/CTS configuration (see Studio *Technical Reference Manual* for valid configurations).
9	Wrong ID (Attach message)	▪ The driver received the wrong attach response.	▪ Check device and Studio Communication Parameters.
11	Invalid Bit Operation	▪ Tried to use Bit with an invalid operand	▪ Check the valid configuration in the address field.
-15	Timeout waiting start a message	▪ Disconnected cables ▪ PLC turned off, or in Stop or error mode ▪ Wrong Station number ▪ Wrong RTS/CTS control settings	▪ Check the cable wiring ▪ Check the PLC state. It must be RUN. ▪ Check the station number. ▪ Check the right configuration. See "**Error! Reference source not found.**" for the different RTS/CTS valid configurations.
-17	Timeout between rx char.	▪ PLC in stop or error mode ▪ Wrong station number ▪ Wrong parity ▪ Wrong RTS/CTS configuration settings	▪ Check the cable wiring ▪ Check the PLC state. It must be RUN ▪ Check the station number. ▪ Check the right configuration. See "**Error! Reference source not found.**" for the different RTS/CTS valid configurations.

⇨ **Tip:**

You can monitor communication status by establishing an event log in Studio's *Output* window (*LogWin* module). To establish a log for **Field Read Commands**, **Field Write Commands** and **Serial Communication,** right-click in the *Output* window and select the desired options from the pop-up menu.

You can also use the *Remote LogWin* module (**Tools** → **Remote LogWin**) to establish an event log on a remote unit that runs Windows CE.

If you are unable to establish communication between Studio and the target device, then try instead to establish communication using the device's own programming software (e.g., Proficy Machine Edition). Quite often, communication is interrupted by a hardware or cable problem or by a device configuration error. If you can successfully communicate using the programming software, then recheck the driver's communication settings in Studio.

To test communication between Studio and the device, we recommend using the sample application provided rather than your new application.

If you must contact us for technical support, please have the following information available:

- **Operating System** (type and version): To find this information, select **Tools** → **System Information**.
- **Project Information**: To find this information, select **Project** → **Status.**
- **Driver Version** and **Communication Log**: Displays in the Studio *Output* window when the driver is running.
- **Device Model** and **Boards**: Consult the hardware manufacturer's documentation for this information.

Sample Application

A sample application that employs the FANUC driver is provided on the Studio installation CD. We strongly recommend that you use this sample application to test the driver *before* you develop your own applications, for the following reasons:

- To better understand the information and instructions provided in this document;
- To verify that your driver configuration is working satisfactorily with the target device; and
- To ensure that the all of hardware used in the test (i.e. the device, adapter, cable, and PC) is functioning safely and correctly.

> **Note:**
> The following instructions assume that you are familiar with developing project applications in Studio. If you are not, then please review the relevant chapters of the Studio *Technical Reference Manual* before proceeding.

To use the sample application:

1. Configure the device's communication settings according to the manufacturer's documentation.
2. Run Studio.
3. From the main menu bar, select **File → Open Project**.
4. Insert the Studio installation CD and browse it to find the sample application. It should be located in the directory **\COMMUNICATION EXAMPLES\FANUC**.
5. Select and open the sample application.
6. Configure and test the driver, as described in the rest of this document.

When you have thoroughly tested the driver with your target device, you may proceed with developing your own Studio application projects.

> **Tip:**
> You can use the sample application screen as the maintenance screen for your own applications.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of Changes
A	2.00	Roberto V. Junior	30-Jul-1999	First driver version Driver available for Windows CE
	2.01	Roberto V. Junior	11-Dec-2000	Included bit reading for AI, AQ and R registers
	2.02	Roberto V. Junior	11-Jan-2001	Included MAIN DRIVER SHEET feature
	2.03	Roberto V. Junior	07-Jan-2003	Increase offset limit of driversheet
	2.04	Lourenço Teodoro and Fabio H.Y.Komura	15-Jul-2003	Configurable BlockSize, AttachTime, and T1 to increase performance Included Float Point to read operations
B	2.05	Fabio H.Y.Komura	24-Jan-2005	Changed the default value of the AttachTime parameter.
C	2.06	Eric Vigiani	28-Jun-2005	Included Attach Time in the message.
D	2.07	Fabio H.Y.Komura	20-Feb-2006	Included Float point to write operations Fixed DWord writing operations Fixed problem with default value for AttachTime
E	2.07	Michael D. Hayden	16-Jun-2006	Edited for language and usability.
F	2.08	Graziane C. Forti	26-Sep-2006	Fixed problem with writing of bit values using trigger in the Standard Driver Sheet
G	2.09	Graziane C. Forti	22-Nov-2006	Fixed problem with writing and reading bit
H	2.10	Eric Vigiani and Graziane C. Forti	21-Feb-2008	▪ Modified to treat correctly the error responses from PLC. ▪ Removed “First Station” field in Communication Parameters dialog.
I	10.1	Marcelo Carvalho	07-Jan-2009	▪ Updated driver version, no changes in the contents.
J	10.3	Fellipe Peternella	01-Jul-2009	▪ Driver modified to allow multiple connections when using Serial Encapsulation over TCP/IP or UDP/IP
K	10.4	Fellipe Peternella Paulo Balbino André Körbes	21-Jul-2010	▪ Improved Write Performance ▪ Allowed bit write using write trigger ▪ Extended the block size for digital headers
L	10.4	Andre Bastos	26-Jan-2011	▪ Doc revision only

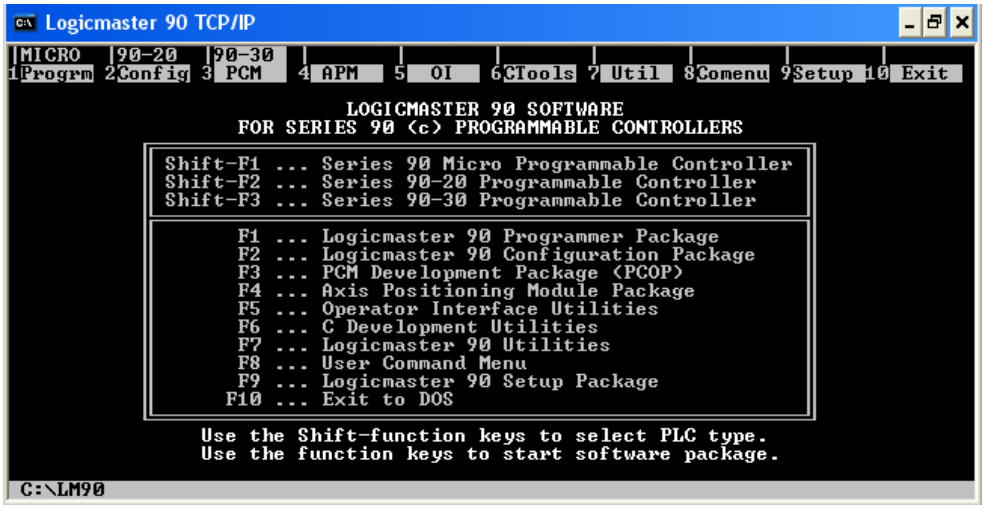
Using Logicmaster 90 Software to configure Communication Parameters

Logicmaster 90 software is a PC-based software package used to configure logic and data access routines for GE Fanuc programmable logic controllers.

Before you try to establish communication with Studio, we recommend using the Logicmaster software to configure some communication parameters through a serial connection.

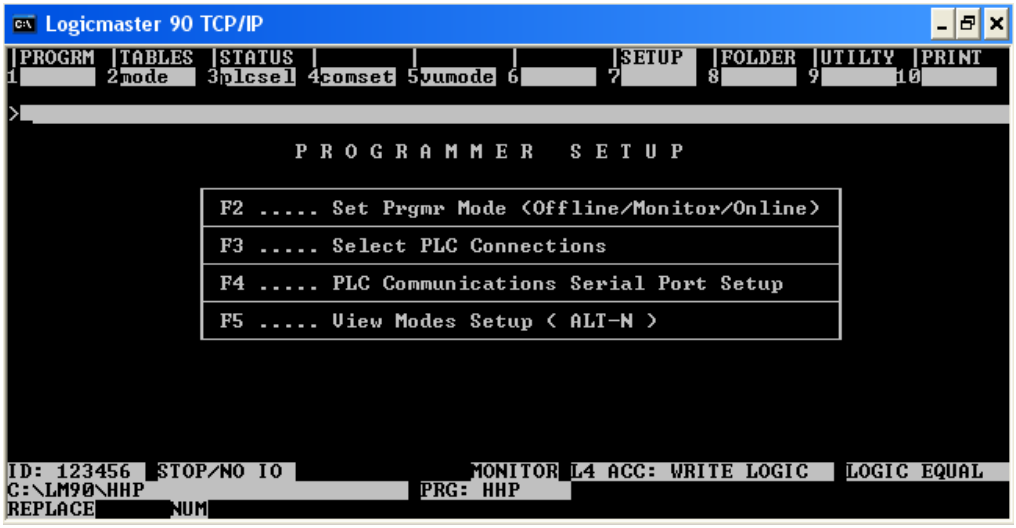
Use the following step-by-step procedure to set up parameters like PLC ID, Baud rate, COM Port, and so on.

1. Connect your computer and the PLC through a serial cable.
2. Start the Logicmaster software.
3. When the Logicmaster application displays, select **Logicmaster 90 Programmer Package** from the main screen by pressing F1 button.

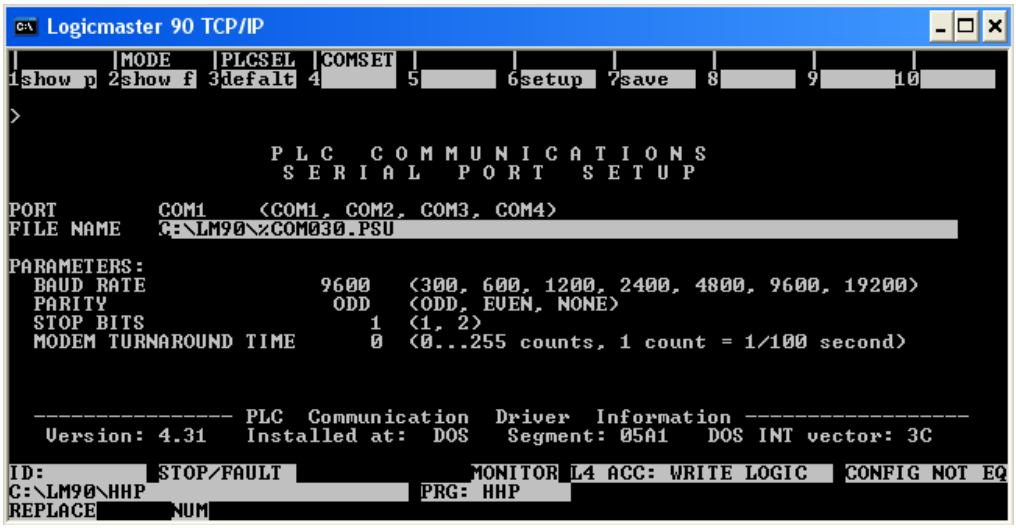


4. The following screen should be displayed:

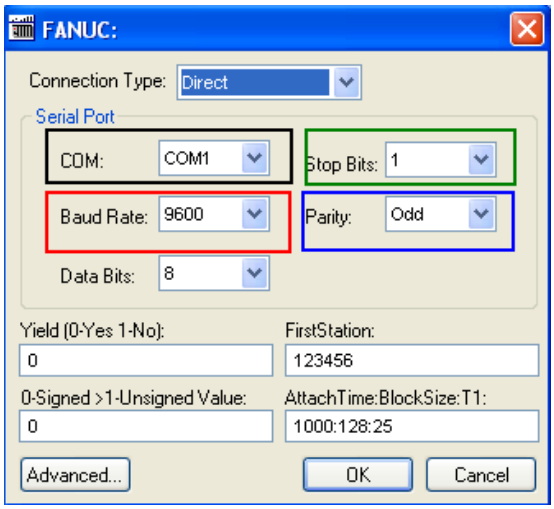
*5. Select **Programmer Mode and Setup** by pressing F7 button. You should get the following screen, as follows below:*



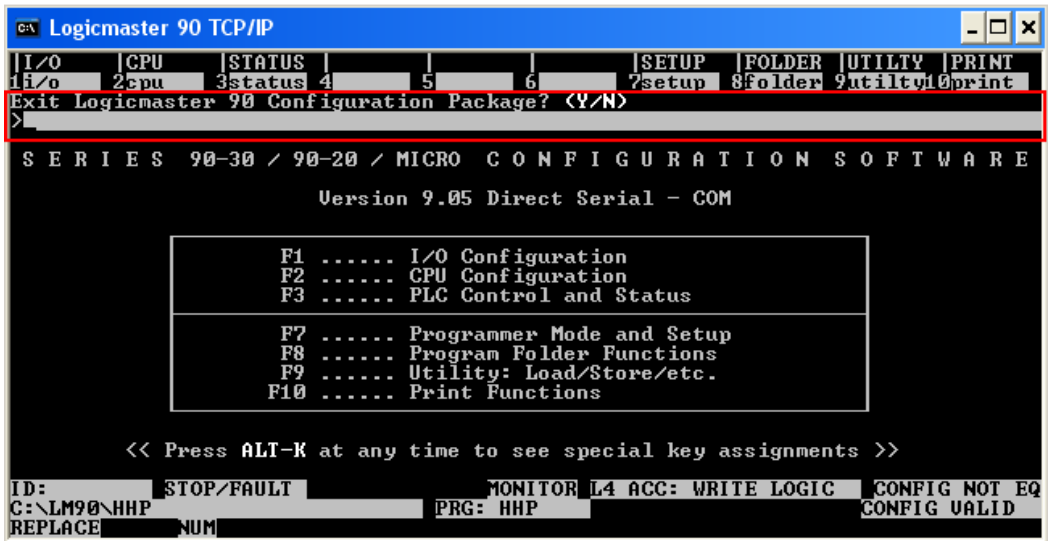
*6. Select **PLC Communications Serial Port Setup** by pressing F4 button. The following screen should be displayed:*



7. To change the default values of these parameters, use the scroll narrows to navigate through the screen and press TAB button to change values. Don't forget to save your changes by pressing F7 button.
Note that these parameters must be configured exactly the same it is configured at Communication Parameters as follows:



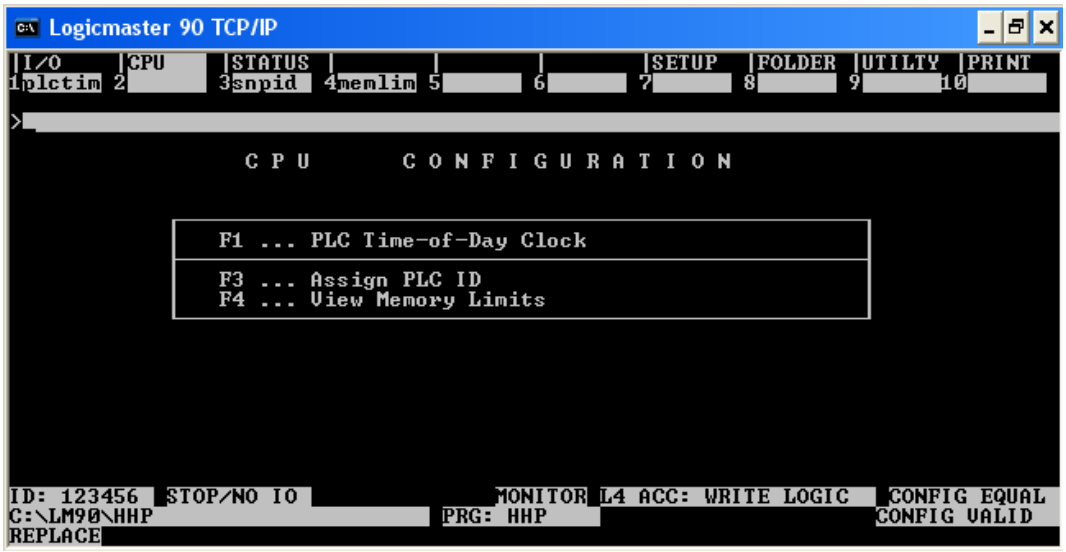
8. Go back to the main screen by pressing ESC three times. The following screen should be displayed:



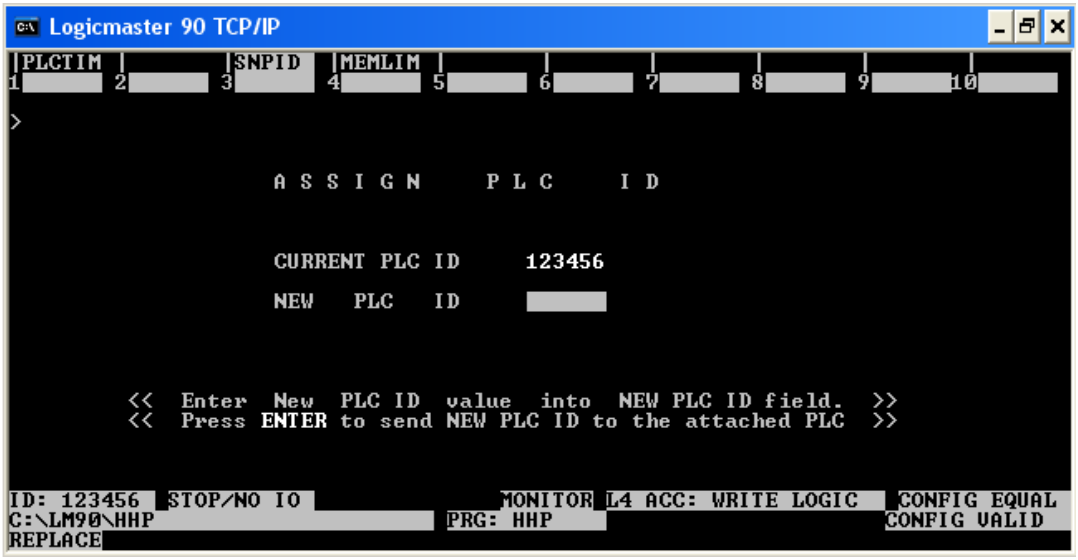
*9. Confirm by pressing **Y** buttom. The main screen should be displayed (Figure 1).*

*10. Select **Logicmaster 90 Configuration Package** by pressing F2 button. The following screen should be displayed:*

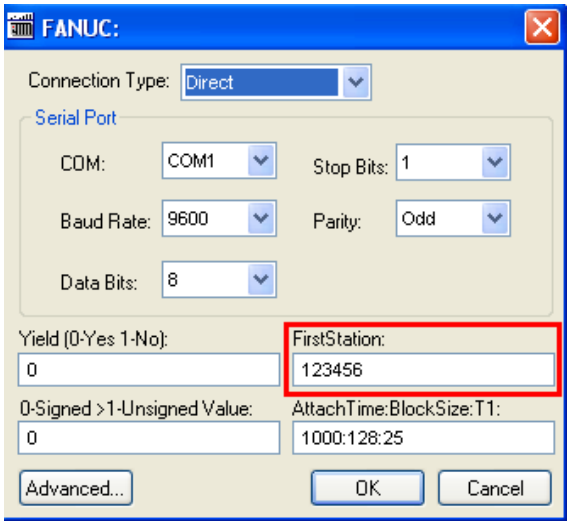
*11. Select **CPU Configuration** by pressing F2:*



*12. Select **Assign PLC ID** by pressing F3. The following screen should be displayed:*



*13. To change the current PLC ID, use the scroll narrows to navigate through the screen until **NEW PLC ID** field and type the desired PLC ID.*
Note that this parameter must be configured exactly the same it is configured at Communication Parameters as follows:



➢ *Other Configuration Parameters:*

- *T1:*

  T1 is the amount of time that must elapse between the reception or transmission of the last character of one message and the transmission of the first character of the next message. It is the minimum amount of time the sending device (master or slave) must wait before transmitting either a message or an acknowledgement. This timer is used by both the master and the slave. T1 time must account for the worst case time for the sending device to turn the link around and set up to be a receiving device, including any modem turn around time. The range for T1 is 5 to 50 milliseconds. It also can be understood as the minimum time the PLC takes to process one request. So, communication problems (as timeout) can be result of the configuration of this parameter with small values.

- *Attach Time:*

  The basic concept to understand the parameter Attach Time is Attach Message. An Attach Message is a master-only message that must be the first message sent after a long break (idleness). Attach Time is the amount of time in which the FANUC driver is forced to send an attach message to avoid loss of communication. So, the smallest this value was, the slowest the driver will work and vice-versa.