ATOS Communication Driver

Driver for Serial Communication
with ATOS Devices Using APR03 Protocol

Contents

Introduction

The ATOS driver enables communication between the Studio system and all ATOS devices that use the APR03 protocol, according to the specifications discussed in this document.

This document will help you to select, configure and execute the ATOS 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 ATOS driver in the Studio system.
- **Configuring the Device**: Describes how the target device must be configured to receive communication from the ATOS driver.
- **Configuring the Driver**: Explains how to configure the ATOS driver in the Studio system, including how to associate database tags with device registers.
- **Executing the Driver**: Explains how to execute the ATOS 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 ATOS driver configuration
- **Revision History**: Provides a log of all changes made to the driver and this documentation.

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 XP/7 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 ATOS driver in Studio and a target ATOS device.

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**: ATOS
- **Compatible Equipment**: all devices compatible with the APR03 protocol

Network Specifications

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

- **Physical Protocol**: Serial Communication
- **Logic Protocol**: APR03
- **Device Runtime Software**: None

Driver Characteristics

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

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

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

- Windows XP/7

The ATOS driver supports the following registers:

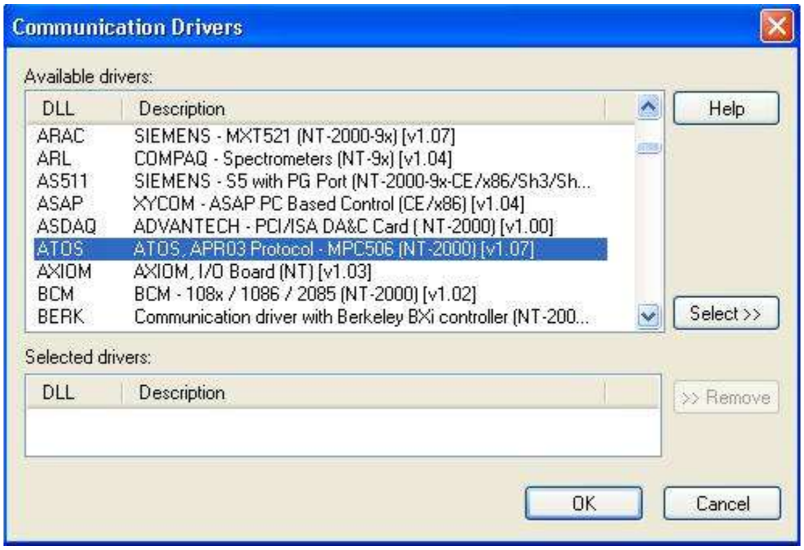
Registers Type	Write	Read	Bit Access	Byte	Word	Dword	String
BIN	•	•	•	•	•	•	–
BCD	•	•	•	•	•	•	–
STRN	•	•	–	–	–	–	•

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 ATOS driver for your Studio application:

1. From the main menu bar, select **Insert** → **Driver** to open the *Communication Drivers* dialog.

2. Select the **ATOS** driver from the *Available Drivers* list, and then click the **Select** button.



Available drivers:

DLL	Description
ARAC	SIEMENS - MXT521 (NT-2000-9x) [v1.07]
ARL	COMPAQ - Spectrometers (NT-9x) [v1.04]
AS511	SIEMENS - S5 with PG Port (NT-2000-9x-CE/x86/Sh3/Sh...
ASAP	XYCOM - ASAP PC Based Control (CE/x86) [v1.04]
ASDAQ	ADVANTECH - PCI/ISA DA&C Card (NT-2000) [v1.00]
ATOS	ATOS, APR03 Protocol - MPC506 (NT-2000) [v1.07]
AXIOM	AXIOM, I/O Board (NT) [v1.03]
BCM	BCM - 108x / 1086 / 2085 (NT-2000) [v1.02]
BERK	Communication driver with Berkeley BXi controller (NT-200...

Selected drivers:

DLL	Description

Communication Drivers Dialog

3. When the **ATOS** 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 ATOS device, you must also install the ATOS programming software. 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 Driver

Once you have selected the ATOS 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.

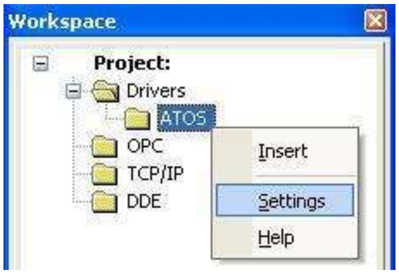
Configuring the Communication Settings

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 ATOS driver-specific settings and procedures will be discussed here. To configure the communication settings for the ATOS driver:

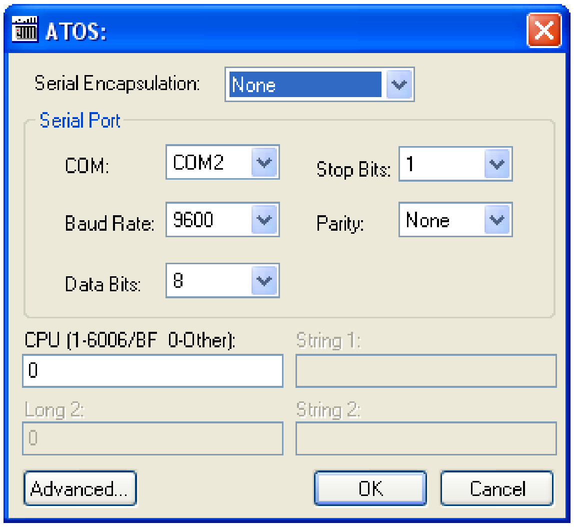
1. In the *Workspace* pane, select the *Comm* tab and then expand the *Drivers* folder. The ATOS driver is listed here as a subfolder.

2. Right-click on the *ATOS* subfolder and then select the **Settings** option from the pop-up menu:



Select Settings from the Pop-Up Menu

The *ATOS: Communication Settings* dialog is displayed:



Serial Encapsulation: None
COM: COM2
Stop Bits: 1
Baud Rate: 9600
Parity: None
Data Bits: 8
CPU (1-6006/BF 0-Other): 0
String 1:
Long 2: 0
String 2:

ATOS: Communication Settings 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.

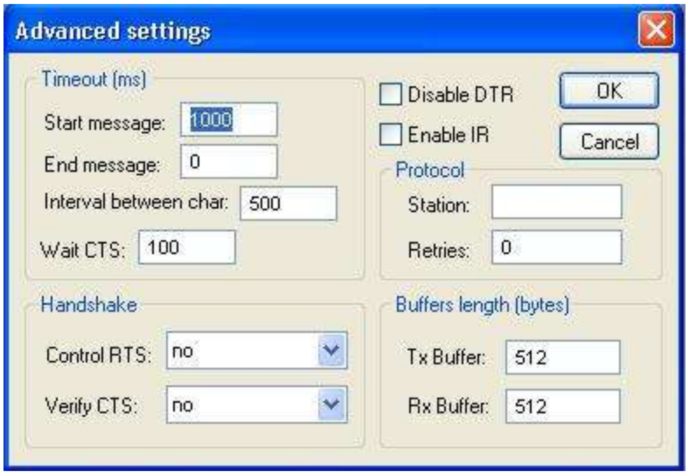
   Depending on your circumstances, you may need to configure the driver *before* you have configured your target device. If this is the case, then take note of the driver settings and have them ready when you later configure the device.

   > ➲ **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:

Parameters	Default Values	Valid Values	Description
CPU (1-6006/BF 0-Other)	**0**	**`0 or 1`**	CPU Type: 1 – 6006/BF CPU 0 – other CPU

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:



Timeout (ms)
Start message: 1000
End message: 0
Interval between char: 500
Wait CTS: 100

□ Disable DTR
□ Enable IR

Protocol
Station:
Retries: 0

Handshake
Control RTS: no
Verify CTS: no

Buffers length (bytes)
Tx Buffer: 512
Rx Buffer: 512

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.
		yes	Set the RTS (Request to Send) handshake signal before communication.
		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 Settings* dialog.

Configuring the Driver Worksheets

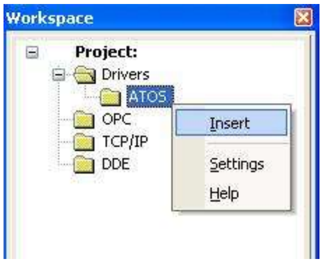
A selected driver includes one or more driver worksheets, which are used to associate database tags in Studio with registers on the target device. Each worksheet is triggered by specific application behavior, so that the tags / registers defined on that worksheet are scanned only when necessary – that is, only when the application is doing something that requires reading from or writing to those specific tags / registers. Doing this optimizes communication and improves system performance.

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.

To insert a new driver worksheet:

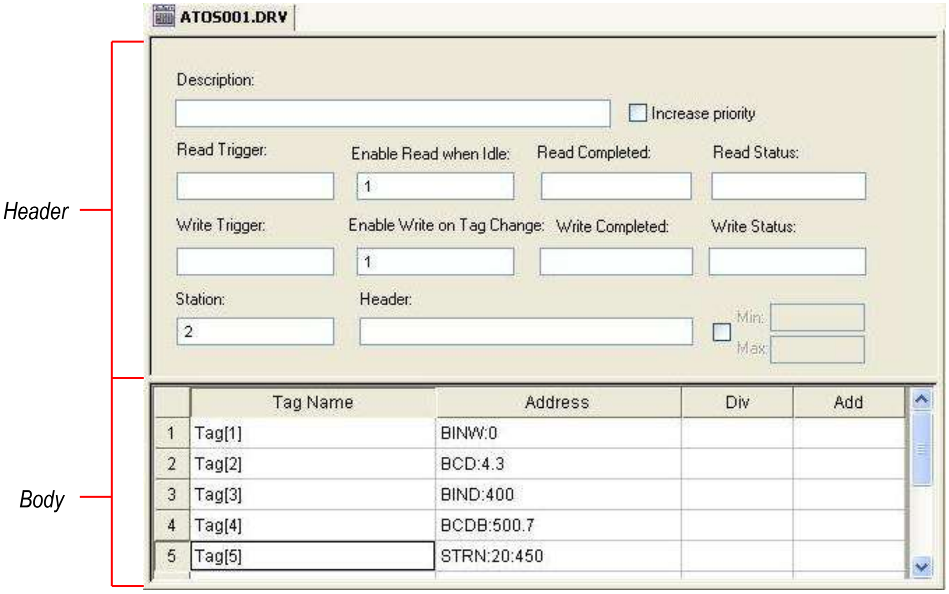
1. In the *Comm* tab, open the *Drivers* folder and locate the *ATOS* subfolder.

2. Right-click on the *ATOS* subfolder, and then select **Insert** from the pop-up menu:



Inserting a New Worksheet

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



Description:
□ Increase priority
Read Trigger:
Enable Read when Idle: 1
Read Completed:
Read Status:
Write Trigger:
Enable Write on Tag Change: 1
Write Completed:
Write Status:
Station: 2
Header:
□ Min: Max:

	Tag Name	Address	Div	Add
1	Tag[1]	BINW:0		
2	Tag[2]	BCD:4.3		
3	Tag[3]	BIND:400		
4	Tag[4]	BCDB:500.7		
5	Tag[5]	STRN:20:450		

ATOS Driver Worksheet

> ✍ **Note:**
>
> Worksheets are numbered in order of creation, so the first worksheet is **`ATOS001.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 ATOS driver.

3. Configure the **Station** and **Header** fields as follows:
   - **Station** field: CLP number on network, from 0 to 31
   - **Header** field: Not used by this driver

4. The **Address** field has an address to be accessed isolated in the PLC, since there is not predicted command blocks.

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

For BIN or BCD format:

`XXXY:N or XXXY:N.b`

Where:
XXX: BCD or BIN
Y: B (byte), W (word), D (double word), F (Flags)
N: Hexadecimal value used to specify the address of the register on the device;

b: optional bit number, must be used with the '.' character
Examples – **`BINW:0, BIND:400, BCD:4.3, BCDB:500.7`**

For Strings:

`ZZZZ:S:N`

Where:
ZZZZ: STRN
S: Used only for the String format, to indicate the length of the String.
N: Hexadecimal value used to specify the address of the register on the device;
Examples – **`STRN:20:450`**

> ✍ **Note:**
>
> The Flags (F) type is special for Boolean areas, where using the Byte (B) type gives values other than True (1) and False (0).

For examples of how register addresses are composed using the values in the **Header** and **Address** fields, see the following table:

Register Address in the Device	Address Field	Format	Type / Length
0	**BINW:0**	BIN	Word
4	**BCD:4.3**	BCD	Bit
400	**BIND:400**	BIN	Double Word
450	**STRN:20:450**	String	20 characters length
500	**BCDB:500.7**	BCD	Bit
1805	**BINF:1805**	BIN	Boolean (Flag)

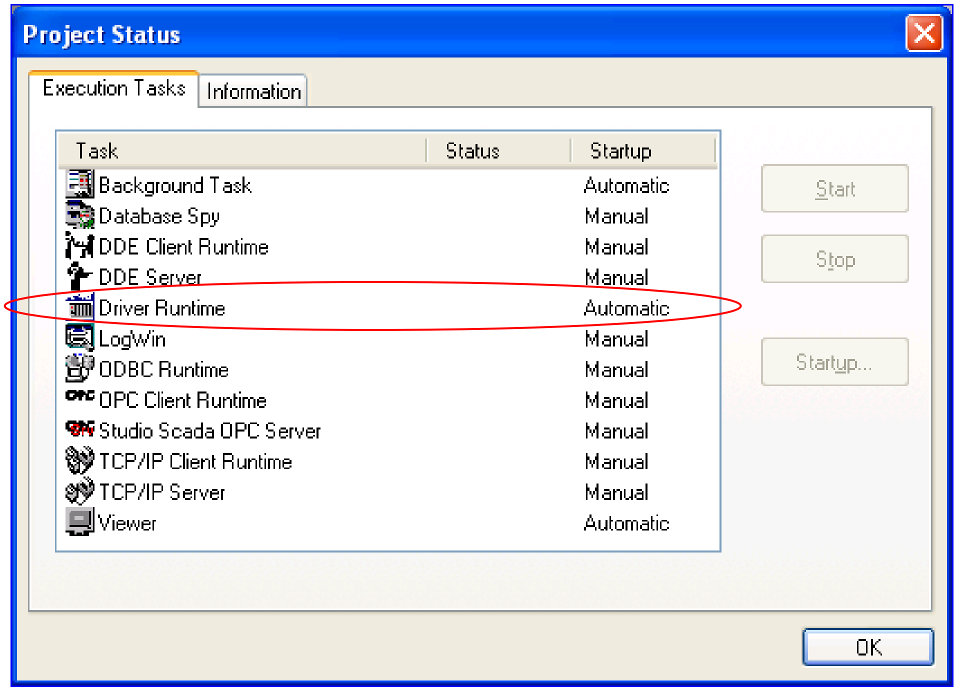
For more information about registers and addressing, please consult the manufacturer's documentation.

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 communication driver(s) will execute correctly:

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



Task	Status	Startup
Background Task		Automatic
Database Spy		Manual
DDE Client Runtime		Manual
DDE Server		Manual
Driver Runtime		Automatic
LogWin		Manual
ODBC Runtime		Manual
OPC Client Runtime		Manual
Studio Scada OPC Server		Manual
TCP/IP Client Runtime		Manual
TCP/IP Server		Manual
Viewer		Automatic

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 ATOS 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 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
3	INVALID ADDRESS	Invalid number in the Address field	Check if the address configured exist into the PLC.
50	INVALID ANSWER	Invalid address for the answer	Check if the address configured exist into the PLC.
60	INVALID CPU	Invalid number in the CPU field of the Communication Parameters	Check if the configured number is a valid CPU number.
-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. Review the Communication Parameters section for valid RTS/CTS configurations.
-17	Timeout between rx characters.	▪ PLC in stop or error mode ▪ Wrong station number ▪ Wrong parity ▪ Wrong RTS/CTS configuration settings	▪ Check the cable wiring ▪ Check the PLC stat e (it must be RUN) ▪ Check the station number ▪ Check the configuration. Review the Communication Parameters section for valid RTS/CTS 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 **Protocol Analyzer,** right-click in the *Output* window and select the desired options from the pop-up menu

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. 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.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of changes
A	1.08	Hugo Tabosa	28-Nov-2006	Modify WORD and DWORD for CPU 4004.05BF
A	1.09	Lourenço Teodoro	16-Dec-2008	Updated driver version, no changes in the contents.
B	1.10	Andre Korbes	14-Jul-2010	Included new format for Boolean areas