



Optimizing ControlLogix Communications

A Guide to Getting the Most from your AB/Rockwell ControlLogix PLC communications

Software Toolbox
International Corporate
Headquarters, USA

148A East Charles Street
Matthews, NC 28105 USA
www.softwaretoolbox.com

TOLL FREE: 888-665-3678
GLOBAL: 704-849-2773
FAX: 704-849-6388



Table of Contents

INTRODUCTION	3
BACKGROUND	4
MAXIMIZING SYSTEM THROUGHPUT	5
RS LINX TO WONDERWARE SPECIFICS	8
HOW TOP SERVER OVERCOMES THE OPC LINK LIMITATION	9



Introduction

The intent of this paper is to guide the control systems engineer or HMI developer in using the Allen-Bradley ControlLogix PLC in the most optimum way for communication to host applications running on a personal computer. Using these suggestions, experienced PLC programmers will improve overall system throughput when using the Software Toolbox TOP Server as the communication software, RS Linx, or any other third party server. PLC Programming methods and other PLC settings will not be discussed in any detail in this paper.

...in many cases, a system of communication is only as good as it's weakest link.

Total system throughput is a function of every component in the system and in many cases, a system of communication is only as good as it's weakest link. This paper takes a systems approach and although it is not exhaustive in coverage of every variable that affects total throughput, it addresses the most important factors for consideration. Any review of system throughput of a user's system should look at the system as a whole, not just one piece, and carefully consider where bottlenecks may be occurring.



Background

First, let's start with a bit of background history. When you communicate with a ControlLogix PLC, the

ControlLogix Multi-Item Request Packet sizes are about 500 bytes... Every character in a tag name takes up about 1 byte.

protocol is not currently as efficient as the older PLC 5 style N7:0 addressing. This is the price you pay for the convenience of tag names in the PLC. If you have ever connected to a PLC 5 or SLC 500 PLC you may recall that you can specify that you want "100 words starting at N7:0" and get them all in one return block. With the ControlLogix PLC you actually have to give it the complete tag name of each tag that you want, the

PLC has to go find that tag in it's database, and return the data to you. When the ControlLogix first came out in late 1998, you could only get one item in each packet, which meant performance was quite slow. Fortunately in early 2000, Rockwell added what they call the CIP (Controller Interface Protocol) Multi-Item Request Packet.

ControlLogix Multi-Item Request Packet sizes are about 500 bytes and can contain the names of multiple PLC tag names that you would like to read. Every character in a tag name (see column in TOP Server tag list called tag address) takes up about 1 byte. There is some amount of these 500 bytes that is needed for the PLC command and general communication overhead. The point remains the same though, with less than 500 bytes available to stuff tag names into a request packet, you're not going to get the "give me 100 words starting here" very easily like you do with the older style PLC 5 addressing.



Maximizing System Throughput

One of the first places to check in your system is the CPU time slice setting in your ControlLogix PLC. This is set using your RS Logix 5000 programming software. This value represents the percentage of available CPU time that is dedicated to communications activities by the PLC CPU. It should be noted though that the time slice settings for CPU communications defaults to 10%. Advantages can be gained by increasing this up to as much as 40%-50%. Values greater than 50% start to reach diminishing returns.

There are different ways of maximizing the communication to the ControlLogix by tag design given the limitations of the PLC protocol. The newest version of RS Linx and associated ControlLogix firmware provide the ultimate in optimizations by using a proprietary pre-mapping system. When this is used RS Linx can perform very well, but it requires higher PLC CPU usage and the start-up connection from RS Linx to the PLC takes significantly longer than solutions that use the CIP Multi-Item Request Packet protocol. Some customers in the field have shared concerns about having to constantly upgrade PLC firmware, RS Linx version, and RS Logix 5000

versions together in order to keep it all working when one of the three is upgraded. Work is currently being done to add similar functionality to the TOP Server as an *option* for those who want to go in this direction – we will still offer the Multi-Item request packet method of communications.

Steps to Maximum Throughput

- 1) Use Global Tags only
- 2) Use Array Tags
- 3) Keep PLC Tag Names Short
- 4) Set PLC CPU time slice to 40%-50%.
- 5) Define Alias Tags with RSLogix

The second, and we are convinced still, the best way to communicate with a ControlLogix is through the use of Arrays and short PLC tag addresses. Arrays may have a long tag address name in the PLC, but if there are a large number of tags in the array great gains in performance can be achieved. This is due to the ability to make a single request for all the tags within the 500-byte packet size. One hundred tags can be read or written to in a single command for example if the Array is of size 100 – i.e. MyTag is an array MyTag[0] to MyTag[99]. Under 200ms times would be reasonable in this situation yet still dependent on the number of other request required to be serviced by the system and demands on the ControlLogix. There is a limit in the ControlLogix CIP protocol on the number of items that can be read from an array in a



single request, and that limit is 3840 word elements. Larger array requests will be broken into the appropriate number of packets by drivers.

The key is that the ControlLogix PLC tag name be as short as possible, not the tag name used in the TOP Server's database

The use of short tag names in the PLC for tags other than arrays is also of great benefit. This is because the TOP Server packs the PLC tag addresses into the Multi-Item Request packet sent to the ControlLogix. The 500-byte limit is what makes the shortness of the tag addresses so critical. Creating all required PC communications tags under the Global file is one way of shortening the names because Global tags require the least amount of space in the Multi-

Item Request Packet. Local (i.e. Program) tags may seem nice, but in the ControlLogix to get at a program tag (vs. a Global) we also have to put in the packet the text "Program:ProgramName." plus the tag name! – You can see how 500 bytes can go fast. Whenever additional structures are used, only and them when this will assist in shortening the tag names more then the name of the structure itself. Since the TOP Server will import L5K file tag descriptions, it is not hard to then change the Server tag names themselves to something more descriptive easing HMI project creation. The key is that the ControlLogix PLC tag name be as short as possible, not the tag name used in the TOP Server's database

Create and use an alias defined in RSLogix5000 for long PLC tag names that simply cannot be avoided, to provide a shorter name for communication. Although this does use some CPU time to update the aliases in the PLC, the gains in communications throughput can outweigh the small price paid in CPU usage. For example, MG1.GroupTaskLoadingFault is a reference to a motion_group type member, where MG1 is the name of the motion_group structure. If this is a highly requested item, create an alias like "mg1_GTLF". This saves 17 bytes per request, room for another request possibly! Using aliases under the Global group can maintain the benefits of the ControlLogix naming convention for internal programming while maximizing communications throughput potential. This opens up the potential for greater Array usage also, which will also improve ControlLogix to ControlLogix (i.e. PLC to PLC) communications. Whenever information from the ControlLogix is requested by PLC tag name, the PLC CPU needs to go through its mapping to determine where in memory the information is in order to get and pass it. Requests of structured-data, including predefined and user-defined data-types, can require additional CPU time to find the information. The only cases where these mapping issues are

Create and use an alias defined in RSLogix5000 for long PLC tag names that simply cannot be avoided



avoided are when a PLC is communicating to a ControlLogix and can only make requests using the standard Allen-Bradley data type tables or using initial start-up mapping as RS Linx currently does.



RS Linx to Wonderware Specifics

While RS Linx has some advantages currently communicating to the ControlLogix, namely it's ability to bypass the limitations of the packet size in the Multi-Item Request Packet, it has disadvantages worth noting when RS Linx is communicating with Wonderware. RS Linx uses OPC interfaces to communicate with Wonderware. While OPC in general is a great way to communicate, Wonderware requires the use of an external application (separate EXE) provided by Wonderware called OPCLink to communicate with OPC. This adds an additional layer of inter-process communication into the system. Inter-process communication is an "computationally expensive" process because of the added memory bounds checking, security checking, and handling of data that occurs in the operating system when crossing from one EXE (i.e. InTouch) to another EXE (i.e. OPC link) to another EXE (i.e. RS Linx). An OPC client passes the requests to OPCLink via OPC connection standards (a subset of MS COM standards). OPCLink then massages the data and creates SuiteLink or FastDDE messages and sends them to Wonderware. Wonderware to OPC Server communications are the same in reverse. This takes some about of time and PC resource usage to do for every request. This system of connecting to an OPC server from InTouch also requires additional set up time when creating or changing a project. For example, you must put a lower case letter designating the data type in front of the item name in InTouch. Contrary to popular belief this is NOT a requirement of OPC servers – this data type indicator is there solely for the benefit of InTouch and OPC Link – the OPC server never even sees that data type descriptor – it is stripped away by OPC Link before OPC link passes the item name to the OPC server. The same issues can also create additional project maintenance costs.



How TOP Server Overcomes the OPC Link Limitation

The TOP Server makes SuiteLink and FastDDE connections an option for use with Wonderware (or other clients that have this ability). While we prefer the use of OPC in general to other connection methods, SuiteLink connections have an advantage in Wonderware because OPCLink and the associated CPU intensive inter-process communication overhead are not needed. All tag and topic creation is done between the TOP Server and Wonderware. The TOP Server also has an Alias map

capability that can give the SuiteLink/FastDDE user some of the group update rate control capabilities of OPC, allowing the user to optimize their performance by only reading the most critical tags at fast scan rates and the rest at scan rates closer to what the "real" application needs. As an other option, the TOP Server has a Tag Generation Wizard that can create Wonderware tags that link directly to the TOP Server's tag database through the Alias map. This tool can significantly decrease the time it takes to create a new project and when using the best methods can speed up system design changes.

The TOP Server also has an Alias map capability that can give the SuiteLink/FastDDE user some of the group update rate control capabilities of OPC.

