ADS Client Specification

FUB “ACL”

Customer: Vistaprint Winterthur

Project: Line ALADDIN

Projectnr.: 10116.04.01

Version: 0.01

Storage: D:\daten\60\_Projekte\10116 Vista Print\04 Line - ALADDIN\01 Development\20\_Engineering\20\_Development\Specification\Line-Aladdin Shirtstream PLC Spec V0.12.docx

# Table of contents

[1 Table of contents 2](#_Toc379522616)

[2 Introduction 3](#_Toc379522617)

[2.1 Scope 3](#_Toc379522618)

[2.2 Reference Documentation 3](#_Toc379522619)

[2.3 Version 3](#_Toc379522620)

[2.4 Abbreviations, definitions, glossary 3](#_Toc379522621)

[3 Common information’s 4](#_Toc379522622)

[3.1 Used software components and libraries 4](#_Toc379522623)

[3.2 Used ADS Index-Group 4](#_Toc379522624)

[3.3 ADS error codes 4](#_Toc379522625)

[3.4 Concept (Example configuration) 5](#_Toc379522626)

[4 Controller Description 6](#_Toc379522627)

[4.1 Level 0 Element: ADS client (Class ACL) 6](#_Toc379522628)

# Introduction

## Scope

This document gives an overview how to use the FUB “ACL” (ADS client) and is solely for the benefit of Vistaprint and all the persons that are involved at platform development. These are software developers, quality engineers and maintenance engineers.

## Reference Documentation

|  |  |  |
| --- | --- | --- |
| Documents | Version | Datum |
| Beckhoff InfoSys - [ADS Index-Group/Offset](http://infosys.beckhoff.com/index.php?content=../content/1031/tcadsdeviceplc/html/tcadsdeviceplc_indexadsservice.htm&id=) |  |  |
| Beckhoff InfoSys - [AdsErrorCode](http://infosys.beckhoff.de/english.php?content=../content/1033/tc3_adsnetref/html/TwinCAT.Ads.AdsErrorCode.html&id=) |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Version

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Author | State | Version | Date |
| Start | AVME/DSC | d | 0.1 | 2014-02-05 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

State: **d** = design, **r** = released

## Abbreviations, definitions, glossary

|  |  |
| --- | --- |
| Designation | Name |
| ADS | "Automation Device Specification” Describes a device and fieldbus-independent interface which controls the communication between ADS device(s). |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Common information’s

## Used software components and libraries

|  |  |  |
| --- | --- | --- |
| Description | Version | Datum |
| TwinCAT system library (Tc2\_System) | V3.3.4.0 | 2013-09-02 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Used ADS Index-Group

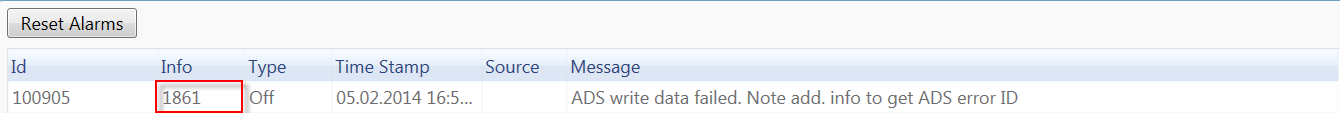
|  |  |
| --- | --- |
| Index | Description |
| 16#F003 | GET\_SYMHANDLE\_BYNAME |
| 16#F005 | READ\_WRITE\_SYMVAL\_BYHANDLE |
|  |  |
|  |  |
|  |  |

## ADS error codes

Follow the link for more details about ADS error codes: [AdsErrorCode](http://infosys.beckhoff.de/english.php?content=../content/1033/tc3_adsnetref/html/TwinCAT.Ads.AdsErrorCode.html&id=)

The error code is an additional information of alarms set by ACL controller.

Example: 1861 (ClientSyncTimeOut)



## Concept (Example configuration)

The following diagram shows an example topology of an ADS network. Over a method called “AddConnection” a new connection can be added to FUB\_ACL. (Note example configuration to port 852)

In ACL Run state, all connections from the list will be polled dependent the configured polling interval.

The internal error handling only check if there are any FUBs errors active. Note: There is no communication watchdog available. A separate data channel must be defined for this purpose if required.



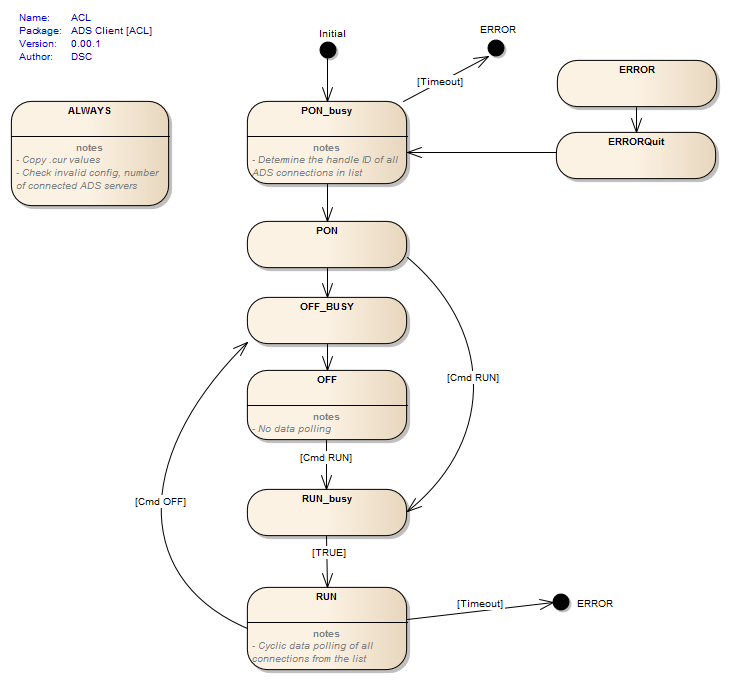
# Controller Description

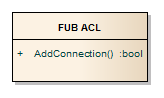
## Level 0 Element: ADS client (Class ACL)

**Functionality**

The ADS client controls the data exchange between various ADS devices. Devices can be added as server connections to that FUB.

**Main Sequence**





**Commands**

* PON (Determines the handle IDs of all connections from the internal list)
* OFF (No data exchange)
* RUN (Data polling enabled)

**Parameter**

|  |  |  |
| --- | --- | --- |
| *Variable* | *Datatype* | *Description* |
| udiTOPowerOn\_ms | UDINT | [ms] Timeout power on. Timeout for initialize all ADS connections. |
| udiTOReadData\_ms | UDINT | [ms] Timeout read data |
| udiTOWriteData\_ms | UDINT | [ms] Timeout write data |
| udiTORWData\_ms | UDINT | [ms] Timeout read/write data |
| udiPollingInterval\_ms | UDINT | [ms] Polling interval for read/write of all ADS connections in list |

**Current Values**

|  |  |  |
| --- | --- | --- |
| *Variable* | *Datatype* | *Description* |
| uinNumOfAdsConnections | UINT | Shows the number of ADS (server) connections |
|  |  |  |
|  |  |  |

**Connection data T\_ADS\_ConnectionData**

|  |  |  |
| --- | --- | --- |
| *Variable* | *Datatype* | *Description* |
| udiConID | UDINT | Connection ID. Automatically assigned by calling method for adding new connection |
| udiHandle | UDINT | Handle ID for data exchange. Determined automatically during state PONbusy. |
| strRoute | STRING | Route of the PRG or FUB instance on server side. Example: 'SlowPRG\_1.fbData.SIf.cur.st\_ADSData' |
| uinTargetPort | UINT | ADS target port |
| strNetID | STRING(30) | Target NetID |
| eAccessType | E\_ADS\_AccessType | Access type |
| udiDataLen | UDINT | Length of the data structure |
| pData | PVOID | Pointer to data structure. Note: Defined structure must be the same on client and server side. |
|  |  |  |

**Errors**

* Initialization error. Minimum one ADS connection must be configured category: OFF
* Timeout initialize ADS connection(s) category: OFF
* Init of ADS connection failed. Note add. info for affected connection ID category: OFF
* ADS read data failed. Note add. info to get ADS error ID category: OFF
* ADS write data failed. Note add. info to get ADS error ID category: OFF
* ADS connection fault. Note add. info for affected connection ID category: OFF

## 