Okuma America Corporation

THINC-API
Release Notes for Lathe

Document No.: S5015-008-25

Revision History

Date	Version	Description	Author
5/21/2007	S5015-008-00	Public release for Lathe THINC-API version 1.0.0.0	LHuynh
6/04/2007	S5015-008-01	Public release for Lathe THINC-API version 1.1.0.0	Lhuynh
08/15/2007	S5015-008-02	Public release for Lathe THINC-API version 1.2.0.0	Lhuynh
2/22/2008	S5015-008-03	Public Release 1.6.0.0 for Lathe THINC-API	Lhuynh
4/11/2008	S5015-008-04	Public Release 1.6.0.0 for Lathe THINC-API	Lhuynh
06/27/2008	S5015-008-05	Public Release 1.6.3.0 for Lathe THINC-API	Lhuynh
07/25/2008	S5015-008-06	Public Release 1.6.4.0 for Lathe THINC-API	Lhuynh
10/27/2008	S5015-008-07	Public Release 1.7.0.0 for Lathe THINC-API	Lhuynh
12/18/2008	S5015-008-08	Public Release 1.8.0.0 for Lathe THINC-API	Lhuynh
04/30/2009	S5015-008-09	Public Release 1.9.1.0 for Lathe THINC-API	Lhuynh
10/12/2009	S5015-008-10	Public Release 1.10.0.0 for Lathe THINC-API	Lhuynh
04/28/2010	S5015-008-11	Public Release 1.11.0.0 for Lathe THINC-API	Lhuynh
09/14/2010	S5015-008-12	Public Release 1.11.1.0 for Lathe THINC-API	Lhuynh
01/04/2011	S5015-008-13	Public Release 1.12.0.0 for Lathe THINC-API	Lhuynh
02/02/2011	S5015-008-14	Public Release 1.12.1.0 for Lathe THINC-API	Lhuynh
11/14/2011	S5015-008-15	Beta Release 1.14.0.0 for Lathe THINC-API	Lhuynh
01/15/2012	S5015-008-16	Beta Release 1.14.1.0 for Lathe THINC-API	Lhuynh
09/21/2012	S5015-008-17	Beta Release 1.14.2.0 for Lathe THINC-API	Lhuynh
12/04/2012	S5015-008-18	Public Release 1.15.0.0 for Lathe THINC-API	Lhuynh

03/06/2013	S5015-008-19	Beta Release 1.15.1.0 for Lathe THINC-API	Lhuynh
03/18/2013	S5015-008-20	Beta Release 1.15.2.0 for Lathe THINC-API	Lhuynh
09/25/2013	S5015-008-21	Public Release 1.16.0.0 for Lathe THINC-API	Lhuynh
01/15/2014	S5015-008-22	Public Release 1.17.0.0 for Lathe THINC-API	Lhuynh
04/01/2014	S5015-008-23	Public Release 1.17.1.0 for Lathe THINC-API	Lhuynh
10/08/2014	S5015-008-24	Public Release 1.17.2.0 for Lathe THINC-API	Lhuynh
10/18/2015	S5015-008-25	Public Release 1.18.0.0 for Lathe THINC-API	Lhuynh

Table of Contents

1. Introduction 5
   1.1 Disclaimer of Warranty 5
   1.2 Purpose 5
   1.3 Scope 5
   1.4 Definitions, Acronyms, and Abbreviations 5
   1.5 References 5

2. About This Release 5

3. Features 7
   3.1 DATA-API 7

4. Known Bugs and Limitations 10
   4.1 Defect 10
   4.2 General Defect 10
      4.2.1 Data-API 10
   4.3 Defects Fixed in this Release 11
      4.3.1 Data-API 11

Release Notes for Lathe

1. Introduction

1.1 Disclaimer of Warranty

Okuma America Corporation makes no representations or warranties, either expressed or implied, by or with respect to anything in this document, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

1.2 Purpose

The purpose of the ***Release Notes*** document is to communicate major new features and changes in this release of the THINC-API for Lathe libraries. It also documents known problems and workarounds.

1.3 Scope

This document describes Public Release 1.18.0.0 of THINC-API for Lathe.

1.4 Definitions, Acronyms, and Abbreviations

GAC – Global Assembly Cache Windows folder located in 'C:\WINDOWS\assembly'

1.5 References

None.

2. About This Release

Public Release of the THINC-API library for Lathe supports the following:

From this release and forward, THINC-API libraries will check dependency libraries during installation. THINC-API will fail to install if version of dependency OCJ libraries cannot support current version of THINC-API.

From this release and forward, API Notifier will delay the checking of API for an approximately of 1 minutes or so after NC is running.

All applications compiled with Beta Release from version 1.15.X.X must be compiled with Public Release version 1.17.0.0 or higher when it is available.

In current version of THINC-API, some of the existing functions related to ATC, Tool, TailStock, and Chuck Data from DATA-API or Command API might not function correctly on OSP-P300S (SLP) and OSP-P300L control unless they have been revised as listed in the Change section 3.2.

Please refer to the help file for detail usage and compatibility information of each function.
This version requires latest OSP system disk.

Version of Okuma.CLDATAPI.dll in this release is 2.7.0.0

Version of Okuma.CLCMDAPI.dll in this release is 2.1.0.0

Version of APINotifierService.exe in this release is 1.17.6.0

Version of APINotifierStatus.exe in this release is 1.1.2.0

Version of Okuma.Flexnet.dll in this release is 1.3.0.0

Version of Okuma.Apilog.dll in this release is 1.4.0.0

This release requires OCJ custom API version 003R on target machine. THINC-API will verify the existing of OCJ custom API version before performing the installation.

The PLC system package listed in the table per control type is also required.

OSP	PLCS package
P100II/P200	From 110A to 110C
P200A Type1	From 120A to 130A
P200A Type2/P300	From 201B to 201G, or 300A and over

The following functions of Okuma.CLDATAPI.dll library will not be available in OSP-P100II control machines:

Classes	Interfaces
CMachine	OnOffStateEnum GetNCStatus(NCStatusEnum enNCStatus)
CMachine	CCurrentAlarm* GetCurrentAlarm()
CIO	OnOffStateEnum GetUserTaskIOVariable(IOTypeEnum enIO, Int32 intIndex) ;
CIO	void SetUserTaskOutputVariable(Int32 intIndex, OnOffStateEnum enValue) ;
CIO	OnOffStateEnum GetProtectedUserTaskOutputVariable(Int32 intIndex) ;
CIO	void SetProtectedUserTaskOutputVariable(Int32 intIndex, OnOffStateEnum enValue) ;

The following functions of Okuma.CLCMDAPI.dll library will not be available in OSP-P100II control machines:

Classes	Interfaces
CATC	Sub RegisterToolPot(ByVal intPotNo As Integer, ByVal intToolNo As Integer, ByVal enSettingToolKind As SettingToolKindEnum, ByVal enSettingToolSize As SettingToolSizeEnum, ByVal enReturnMagazine As ReturnMagazineEnum)
CATC	Sub SetNextTool(ByVal intToolNo As Integer, ByVal enSettingToolKind As SettingToolKindEnum, ByVal enSettingToolSize As SettingToolSizeEnum, ByVal enReturnMagazine As ReturnMagazineEnum)
CATC	Sub SetToolInStation(ByVal intToolNo As Integer, ByVal enSettingToolKind As SettingToolKindEnum, ByVal enSettingToolSize As SettingToolSizeEnum, ByVal enReturnMagazine As ReturnMagazineEnum, ByVal enTurretStation As TurretStationEnum)
CATC	Sub UnRegisterToolPot(ByVal intPotNo As Integer)
CProgram	CancelMainProgram()

CProgram	SelectMainProgramRSide(ByVal strMainProgramFileName As String, Optional ByVal strSubProgramFileName As String = "", Optional ByVal strSystemSubtituteProgramFileName As String = "", Optional ByVal strProgramName As String = "")
CProgram	SelectMainProgramLSide(ByVal strMainProgramFileName As String, Optional ByVal strSubProgramFileName As String = "", Optional ByVal strSystemSubtituteProgramFileName As String = "", Optional ByVal strProgramName As String = "")
CTools	CalcualteToolOffset(ByVal intToolNo As Integer, ByVal enAxisIndex As OffsetAxisIndexEnum, ByVal enSubSystem As SubSystemEnum, ByVal dblValue As Double)
CTools	AddConstantToolOffset(ByVal intOffsetNo As Integer, ByVal enAxisIndex As OffsetAxisIndexEnum, ByVal enSubSystem As SubSystemEnum, ByVal enCuttingPosition As CuttingPositionEnum)
CTools	AddConstantNoseRadiusCompensation(ByVal intOffsetNo As Integer, ByVal enAxisIndex As OffsetAxisIndex2Enum, ByVal enSubSystem As SubSystemEnum, ByVal enCuttingPosition As CuttingPositionEnum)
CTools	AddConstantToolWear(ByVal intOffsetNo As Integer, ByVal enAxisIndex As OffsetAxisIndex2Enum, ByVal enSubSystem As SubSystemEnum, ByVal enCuttingPosition As CuttingPositionEnum)
CTools	SubtractConstantToolOffset(ByVal intOffsetNo As Integer, ByVal enAxisIndex As OffsetAxisIndexEnum, ByVal enSubSystem As SubSystemEnum, ByVal enCuttingPosition As CuttingPositionEnum)
CTools	SubtractConstantNoseRadiusCompensation(ByVal intOffsetNo As Integer, ByVal enAxisIndex As OffsetAxisIndex2Enum, ByVal enSubSystem As SubSystemEnum, ByVal enCuttingPosition As CuttingPositionEnum)
CTools	SubtractConstantToolWear(ByVal intOffsetNo As Integer, ByVal enAxisIndex As OffsetAxisIndex2Enum, ByVal enSubSystem As SubSystemEnum, ByVal enCuttingPosition As CuttingPositionEnum)
CProgram	SelectScheduleProgramLSide(ByVal strScheduleProgramFileName As String)
CProgram	SelectScheduleProgramRSide(ByVal strScheduleProgramFileName As String)
CMachine	Public Sub SetUserAlarm(ByVal enAlarm As UserAlarmEnum, Optional ByVal strAlarmMessage As String = "", Optional ByVal enUserAlarmSubSystem As UserAlarmSubSystemEnum = 0)
CMachine	Public Sub ClearUserAlarmD(ByVal enUserAlarmSubSystem As UserAlarmSubSystemEnum)

Note: User Alarm will require OKUMA.Lathe.UserAlarm license feature in order to function on OSP-P200 machine if machine can support.

3. Features

3.1 DATA-API

STATE	Control	Classes	Interfaces
REVISED	P100II P200 P300S	CATC	GetMagazinePosition(ByVal intMagazineNo As Integer) As Integer

REVISED	P100II P200 P300S	CATC	GetMagazineProperty() As MagazineEnum
NEW	P100II P200 P300L P300s	CIO	GetIO(ByVal strLabel As String) As CIOAddress
NEW	P100II P200 P300L P300s	CIO	GetLabel(ByVal objIOAddress As CIOAddress) As String
NEW	P300S	CTools	GetToolAdjustment(ByVal intToolNo As Integer, ByVal enAxisIndex As ToolOffsetAxisIndexEnum, ByVal enEdgeNo As ToolEdgeEnum, ByVal enToolCuttingPosition As ToolCuttingPositionEnum) As Double
NEW	P300S	CTools	SetToolAdjustment(ByVal intToolNo As Integer, ByVal enAxisIndex As ToolOffsetAxisIndexEnum, ByVal enEdgeNo As ToolEdgeEnum, ByVal enToolCuttingPosition As ToolCuttingPositionEnum, ByVal dblValue As Double)
NEW	P300S	CTools	AddToolAdjustment(ByVal intToolNo As Integer, ByVal enAxisIndex As ToolOffsetAxisIndexEnum, ByVal enEdgeNo As ToolEdgeEnum, ByVal enToolCuttingPosition As ToolCuttingPositionEnum, ByVal dblValue As Double)
NEW	P300S	CTools	CalToolAdjustment(ByVal intToolNo As Integer, ByVal enAxisIndex As ToolOffsetAxisIndexEnum, ByVal enEdgeNo As ToolEdgeEnum, ByVal enToolCuttingPosition As ToolCuttingPositionEnum, ByVal dblValue As Double)
NEW	P300L P300S	CTools	GetToolList(ByVal enToolListType As ToolListTypeEnum) As Integer()
NEW	P300L P300S	CTools	GetToolType(ByVal intToolNo As Integer, ByVal enToolEdgeNo As ToolEdgeEnum) As ToolTypeEnum
NEW	P300L P300S	CTools	SetToolType(ByVal intToolNo As Integer, ByVal enToolEdgeNo As ToolEdgeEnum, ByVal enToolType As ToolTypeEnum)

NEW	P300L P300S	CTools	GetToolComment(ByVal intToolNo As Integer) As String
NEW	P300L P300S	CTools	SetToolComment(ByVal intToolNo As Integer, ByVal strComment As String)
NEW	P300L	CTools	GetReferenceToolOffset1(ByVal intToolNo As Integer, _ ByVal enEdgeNo As ToolEdgeEnum, ByVal enSpindleAxisMode As SpindleAxisModeEnum) As Integer
NEW	P300L	CTools	GetReferenceToolOffset2(ByVal intToolNo As Integer, _ ByVal enEdgeNo As ToolEdgeEnum, ByVal enSpindleAxisMode As SpindleAxisModeEnum) As Integer
NEW	P300L	CTools	SetReferenceToolOffset1(ByVal intToolNo As Integer, ByVal enEdgeNo As ToolEdgeEnum, ByVal enSpindleAxisMode As SpindleAxisModeEnum, ByVal intOffsetIndex As Integer)
NEW	P300L	CTools	SetReferenceToolOffset2(ByVal intToolNo As Integer, ByVal enEdgeNo As ToolEdgeEnum, ByVal enSpindleAxisMode As SpindleAxisModeEnum, ByVal intOffsetIndex As Integer)
NEW	P100II P200 P300S	CWorkpiece	GetMaxZeroOffset() As Integer
NEW	P100II P200 P300S	CWorkpiece	GetZeroOffset(ByVal intOffsetIndex As Integer, ByVal enAxisIndex As AxisIndex7Enum) As Double
NEW	P100II P200 P300S	CWorkpiece	SetZeroOffset(ByVal intOffsetIndex As Integer, ByVal enAxisIndex As AxisIndex7Enum, ByVal dblValue As Double)
NEW	P100II P200 P300S	CWorkpiece	AddZeroOffset(ByVal intOffsetIndex As Integer, ByVal enAxisIndex As AxisIndex7Enum, ByVal dblValue As Double)
NEW	P100II P200 P300S	CWorkpiece	CalZeroOffset(ByVal intOffsetIndex As Integer, ByVal enAxisIndex As AxisIndex7Enum, ByVal dblValue As Double)

Please refer to the help file for more information.

4. Known Bugs and Limitations

This section identifies known problems in this release and describes any work-arounds.

4.1 Defect

4.2 General Defect

DATA-API library, Okuma.CLDATAPI.dll, cannot create directly under ASP.NET web application.

Solution/Work-arounds: Create and initialize DATA-API in a separate thread. All function calls must be called from objects created inside separated thread mentioned above.

DATA-API library can only support applications designed with single-threaded apartment of COM threading model. The underlying library, LDATAPI.dll, cannot be loaded during call to CMachine::Init function when an MTAThread attribute is applied to the application.

Solution/Work-arounds: None

4.2.1 Data-API

4.2.1.1 MacMan.COperationHistory class

Function:

Int32 COperation* GetOperationHistory(Int32 intIndex) ;
ArrayList* GetOperationHistory(Int32 intFromIndex, Int32 intToIndex);
Int32 GetMaxCount() ;
Int32 GetCount() ;

Symptom: Failed to get correct data for Subsystem L and R side if MacMan screen is different than current setting of subsystem. It always gets the data from current MacMan screen.

Solution/Work around: None

4.2.1.2 CAxis class

Function:

Double GetActualPositionProgramCoord(AxisIndex1Enum enAxisIndex);
Double GetTargetPosition (AxisIndex1Enum enAxisIndex);

Symptom: The Z-axis position for sub system NC-AL and NC-AR does not return data correctly. It is based on the current selection of spindle in NC panel or command program G140/G141.

Solution/Work around: None

4.3 Defects Fixed in this Release

4.3.1 Data-API

4.3.1.1 CSpindle class

Function:

Public Function GetSpindlerateOverride(ByVal *enAxisIndex* As SpindleAxisIndexEnum) As Integer

Symptom: The function cannot return correct spindle rate override of spindle1/spindle2 on P300 controls.