Okuma America Corporation

THINC-API
Release Notes for Lathe

Document No.: S5015-008-32

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
10/18/2016	S5015-008-26	Public Release 1.19.0.0 for Lathe THINC-API	Lhuynh
12/07/2017	S5015-008-27	Public Release 1.20.0.0 for Lathe THINC-API	Lhuynh
06/14/2018	S5015-008-28	Public Release 1.21.1.0 for Lathe THINC-API	Lhuynh
03/11/2019	S5015-008-29	Public release 1.22.0.0 for Lathe THINC-API	Lhuynh
02/08/2020	S5015-008-30	Public release 1.23.0.0 for Lathe THINC-API	Lhuynh
08/017/2020	S5015-008-31	Public release 1.23.1.0 for Lathe THINC-API	Lhuynh
02/02/2023	S5015-008-32	Public release 1.24.0.0 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 8
   3.1 Command-API 8
      3.1.1 CTools 8
   3.2 Data-API 8
      3.2.1 CTools 8

4. Known Bugs and Limitations 9
   4.1 Defect 9
      4.1.1 General Defect 9
      4.1.2 Firebird Database Log File 9
      4.1.3 Data-API 10
   4.2 Defects Fixed in this Release 10
      4.2.1 Lathe 10

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 the Public release version 1.24.0.0 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

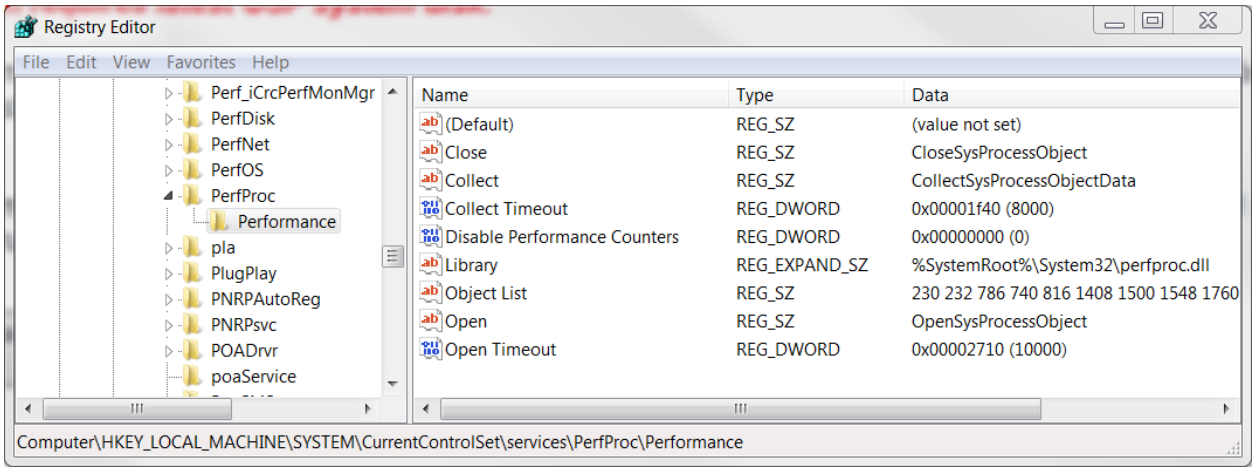
All applications compiled with Beta Release from version 1.15.X.X must be compiled with Public Release version 1.17.0.0 or higher when it is available.

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.

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

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. API Notifier service does use Windows Performance Counters service. As a result, API Notifier will not run correctly if Windows Performance Counters service is not enable. During the installation of THINC-API, Windows Performance Counters will be reset to 0 in the Windows system registry as shown below:



Libraries included in this release for Lathe are compiled with .NET Framework 4.0:

Version of Okuma.CLDATAPI.dll in this release is 3.4.1.0

Version of Okuma.CLCMDAPI.dll in this release is 3.4.0.0

Version of APINotifierService.exe in this release is 1.24.0.0

Version of APINotifierStatus.exe in this release is 1.3.0.0

Version of Okuma.Flexnet.Net4.dll in this release is 1.0.0.0

Version of Okuma.Apilog2.dll in this release is 1.2.0.0

This release requires OCJ custom API version 004B 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

Important: All applications designed with THINC-API libraries must use THINC-API version 1.21.1.0 or higher to be able to run on Microsoft Windows 10.

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 or newer controls if machine can support.

3. Features

Please refer to the help file for detail information of each function.

3.1 Command-API

3.1.1 CTools

Functions	Description
InitToolLife	Initialization of the Tool Life Management Tables After the tool group numbers have been designated for tool numbers and tool offset numbers and the life conditions have been set, the tool life management tables must be initialized. Tool life management tables initialization clears the current number of machined workpieces, time, tool wear amount and the NG flag in tool gauging NG. It also returns the "tool life flag" and "tool group life flag" statuses to "OK". The tool wear amount of the tool offset no. set to the group is also cleared, and the tool number is set as the smallest number registered in the group. After changing all the tools in a setup change, it is essential to initialize the tool life management tables. Notes: Tool wear offset is cleared at tool life management initialization. The tool wear clearing setting at group initialization can be changed by setting the following parameter: Optional Parameter (Long, Word, Bit): No.6 Bit 7 0: Cleared 1: Not cleared Default setting is 0 or cleared at initialization

3.2 Data-API

3.2.1 CTools

Functions	Description
GetToolAdjustmentWidth	For OSP-P300S: Gets tool adjustment width given tool number for current sub system.
	This data field is only visible on NC-HMI when the NC Optional Parameter Bit at [56,0] = 0 for grooving tool
GetToolWearWidth	For OSP-P300S: Gets tool wear width given tool number for current sub system.

	This data field is only visible on NC-HMI when the NC Optional Parameter Bit at [56,0] = 1 for grooving tool
GetToolWeight	For OSP-P300S: Gets tool weight for current sub system.
GetToolOffsetWidth	For OSP-P300S: Gets tool offset width given tool number for current sub system.
	Note:
	Optional Parameter Bit [56,0] = OFF: The value returned from this function will have the same value of Tool Width showing in the Tool Edit page
	Optional Parameter Bit [56,0] = ON: The value returned from this function will be doubled the value of Tool Width showing in the Tool Edit page
IsHighPressureCoolantEnable	For OSP-P300S: Check if a high pressure coolant command is enable or not given the tool number for the current sub system.
IsToolInUse	For OSP-P300: Get tool in use status for tool number, edge number, and tool cutting position for current sub system.

4. Known Bugs and Limitations

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

NOTES: Threading

All THINC-API objects must be created and called from STA threads ONLY. Threads created by Window System such as thread pool which are under MTA threads cannot be supported.

4.1 Defect

4.1.1 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

4.1.2 Firebird Database Log File

THINC-API libraries compiled with .NET Framework 1.1 use Firebird Database engine. The log file created by Firebird database engine can be corrupted, unexpectedly.

Solution/Work-arounds: A bat file is created during the installation of THINC-API and will be executed every time Windows is first started to replace the existing log.fdb located in “D:\Program files\Okuma\LoggingService” to prevent file

corruption.

New THINC-API libraries that compiled with .NET Framework 4.0 will use SQLite database for its logging service. All applications should use new version THINC-API if possible to prevent using Firebird database and to be compatible with Windows 10.

4.1.3 Data-API

4.1.3.1 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

4.1.3.2 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

4.2 Defects Fixed in this Release

4.2.1 Lathe

4.2.1.1 BA Angle

Sympton: BA angle data is reported incorrect value