Okuma America Corporation

THINC-API
Release Notes for Lathe

Document No.: S5015-008-11

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

Table of Contents

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.

Copyright © 2010, Okuma America Corporation.
All rights reserved.

GOVERNMENT RIGHTS LEGEND: Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the applicable Okuma America Corporation license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14, as applicable.

"Okuma America Corporation" and Okuma America Corporation's products are trademarks of Okuma America Corporation. References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

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 1.11.0.0 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 1.11.0.0 of the THINC-API library for Lathe supports the following:

From this release, THINC API libraries will check dependency libraries during installation and at run-time. THINC API will fail to install or load if version of dependency libraries cannot support current version of THINC API.

Version of Okuma.CLDATAPI.dll in Public Release 1.11.0.0 is 2.1.0.0

Version of Okuma.CLCMDAPI.dll in Public Release 1.11.0.0 is 1.7.0.0

Version of APINotifierService.exe in Public Release 1.11.0.0 is 1.11.0.0

Version of Okuma.Flexnet.dll in Public Release 1.11.0.0 is 1.3.0.0

Version of Okuma.Apilog.dll in Public Release 1.11.0.0 is 1.1.0.0

The following functions of Okuma.CLDATAPI.dll library will be only available in OSP-P200 control machines:

Classes	Interfaces
CMachine	OnOffStateEnum GetNCStatus(NCStatusEnum enNCStatus)
CMachine	CCurrentAlarm* GetCurrentAlarm()

The following functions of Okuma.CLCMDAPI.dll library will be only available in OSP-P200 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)

<mark>CProgram</mark>	<mark>SelectScheduleProgramRSide(ByVal strScheduleProgramFileName As String)</mark>

3. New Features

The new classes/functions available in Public Release 1.11.0.0 are:

3.1 Data API

Classes	Interfaces
CAxis	Double GetActualPositionCommandCoord (AxisIndex2Enum enAxisIndex)
CMachine	Int32 GetHourMeterCount(HourMeterEnum enValue) ; System::Void SetHourMeterCount(HourMeterEnum enValue, Int32 intValue) ; System::Void AddHourMeterCount(HourMeterEnum enValue, Int32 intValue) ;
CMachine	Int32 GetHourMeterSet(HourMeterEnum enValue) ; SetHourMeterSet(HourMeterEnum enValue, Int32 intValue) ; System::Void AddHourMeterSet(HourMeterEnum enValue, Int32 intValue) ;
CMachine	MachineSideEnum GetMachineSelection()
CMachine	String GetCurrentAlarm(MachineSideEnum enMachineSide)
CProgram	ProgramRunningStateEnum GetProgramRunningState(MachineSideEnum enMachineSide)
CProgram	String* GetActiveScheduleProgramFileName(MachineSideEnum enMachineSide)
CSpindle	SpindleSelectionEnum GetCurrentSpindleSelection()
CTools	ToolPositionEnum GetToolCuttingPosition()
CWorkpiece	Int32 GetWorkpieceCounterSet(WorkpieceCounterEnum enValue) ; void SetWorkpieceCounterSet(WorkpieceCounterEnum enValue, Int32 intValue) ; void AddWorkpieceCounterSet(WorkpieceCounterEnum enValue, Int32 intValue) ;

3.2 Change

3.2.1 Data API

3.2.1.1 CProgram class

The functions listed below can now support sub system.

Functions:
    String* GetActiveProgramName()
    String* GetExecutedSequenceNumber()

String* GetActiveProgramFileName()

4. Known Bugs and Limitations

This section identifies known problems with Public Release 1.11.0.0 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   API Notifier

Sympton:
    API Notifier application consumes above 90% of CPU when license is expired and NC is running.

The correction of this issue requires using of API Notifier Service version 1.11.0.0 and Okuma Flexnet library version 1.3.0.0.

4.3.2 CProgram class

Function:

    Public Function GetRunningProgram(ByVal *intRow* As Integer,
                                          ByVal *intColumn* As Integer,
                                          ByRef *pintReadPoint* As Integer,
                                          ByRef *pintExecutePoint* As Integer) As String()

Symptom: The function creates memory leak over long run.