

---

**Okuma America Corporation**

---

**THINC-API**  
**Release Notes For Machining Center**

**Document No.: S5015-003-28**

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

## Revision History

Date	Version	Description	Author
5/09/2007	S5015-003-00	Public release for Machining Center THINC-API Release 1.0.0.0	LHuynh
5/11/2007	S5015-003-01	Revise Public release for Machining Center THINC-API Release 1.0.0.0: <ul style="list-style-type: none"> <li>- Remove section 3.0 Compatible and put in Installation Manual document</li> <li>- Remove section 4.0 Upgrades and put in THINC-API Help file.</li> <li>- Revise section 6.1 and 6.2</li> <li>- Use <a href="mailto:api@okuma.com">api@okuma.com</a> email instead of p100issues@okuma.com</li> </ul>	LHuynh
5/21/2007	S5015-003-02	Revise General defect – Cannot support 0.1 Micron option. Revise 4.1 section	LHuynh
06/04/2007	S5015-003-03	Public release for Machining Center THINC-API Release 1.1.0.0	LHuynh
10/19/2007	S5015-003-04	Public release for Machining Center THINC-API Release 1.3.0.0	LHuynh
2/22/2008	S5015-003-05	Public release for Machining Center THINC-API Release 1.6.0.0	LHuynh
04/11/2008	S5015-003-06	Public release for Machining Center THINC-API Release 1.6.0.0	LHuynh
06/27/2008	S5015-003-07	Public release for Machining Center THINC-API Release 1.6.3.0	LHuynh
07/25/2008	S5015-003-08	Public release for Machining Center THINC-API Release 1.6.4.0	LHuynh
10/27/2008	S5015-003-09	Public release for Machining Center THINC-API Release 1.7.00	LHuynh
12/18/2008	S5015-003-010	Public release for Machining Center THINC-API Release 1.8.0.0	LHuynh
04/30/2009	S5015-003-011	Public release for Machining Center THINC-API Release 1.9.1.0	LHuynh
10/12/2009	S5015-003-012	Public release for Machining Center THINC-API Release 1.10.0.0	LHuynh
4/28/2010	S5015-003-013	Public release for Machining Center THINC-API Release 1.11.0.0	LHuynh
09/15/2010	S5015-003-014	Public release for Machining Center	LHuynh

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

		THINC-API Release 1.11.1.0	
12/15/2010	S5015-003-015	Public release for Machining Center THINC-API Release 1.12.0.0	LHuynh
02/02/2011	S5015-003-016	Public release for Machining Center THINC-API Release 1.12.1.0	LHuynh
11/16/2011	S5015-003-017	Beta release for Machining Center THINC-API Release 1.14.0.0	LHuynh
01/15/2012	S5015-003-018	Beta release for Machining Center THINC-API Release 1.14.1.0	LHuynh
09/21/2012	S5015-003-019	Beta release for Machining Center THINC- API Release 1.14.2.0	LHuynh
12/04//2012	S5015-003-020	Public release for Machining Center THINC-API Release 1.15.0.0	LHuynh
04/19/2013	S5015-003-021	Beta release for Machining Center THINC-API Release 1.15.3.0	LHuynh
09/25/2013	S5015-003-022	Public release for Machining Center THINC-API Release 1.16.0.0	ASlagle
01/27/2014	S5015-003-023	Public release for Machining Center THINC-API Release 1.17.0.0	LHuynh
04/02/2014	S5015-003-024	Public release for Machining Center THINC-API Release 1.17.1.0	LHuynh
10/08/2014	S5015-003-025	Public release for Machining Center THINC-API Release 1.17.2.0	LHuynh
10/18/2015	S5015-003-026	Public release for Machining Center THINC-API Release 1.18.0.0	LHuynh
10/18/2016	S5015-003-027	Public release for Machining Center THINC-API Release 1.19.0.0	LHuynh
12/07/2017	S5015-003-028	Public release for Machining Center THINC-API Release 1.20.0.0	LHuynh

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

## 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	10
3.1	DATA-API	10
3.2	Command API	10
4.	Known Bugs and Limitations	11
4.1	Defect	11
4.1.1	Firebird Database Log File	11
4.1.2	Data-API	11
4.2	Defects Fixed in this Release	12

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

# Release Notes For Machining Center

## 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 © 2017, 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 Machining Center libraries. It also documents known problems and workarounds.

### 1.3 Scope

This document describes Release 1.20.0.0 of THINC-API.

### 1.4 Definitions, Acronyms, and Abbreviations

GAC – Global Assembly Cache Windows

### 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 1.16.0.0 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 (MP) and OSP-P300M control.***

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

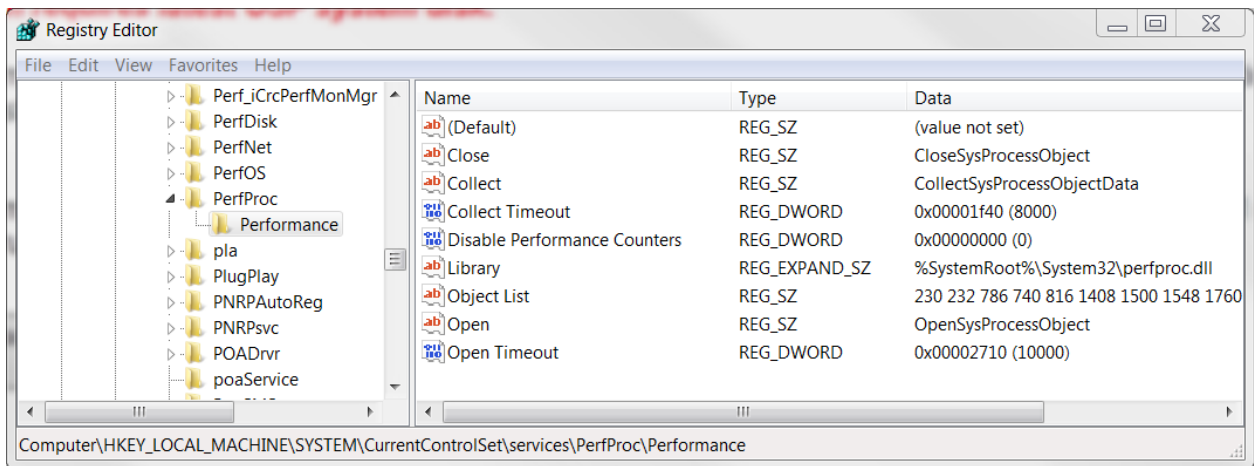
From this release and forward, THINC API libraries will check dependency libraries during installation. THINC API will fail to install if version of dependency OCJ libraries cannot support current version of THINC API.

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

THINC-API Release notes	©Okuma America Corporation, 2017	Page 5 of 12
-------------------------	-------------------------------------	--------------

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

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 Machining Center are compiled with .NET Framework 4.0:

Version of Okuma.CMDATAPI.dll in this release is 3.0.0.0

Version of Okuma.CMCMDAPI.dll in this release is 3.1.0.0

Version of APINotifierService.exe in this release is 1.20.0.0

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 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.CMDATAPI.dll library will be only available based on the version of NC control software:

Class	Interfaces
CTools2	Int32 GetMaxTools()
CTools2	Int32 GetToolNo(Int32 intPotNo)

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

CTools2	Int32 GetMaxPots()
CTools2	String* GetGroupNo(Int32 intPotNo)
CTools2	Int32 GetSerialNo(Int32 intPotNo)
CTools2	String* GetToolName(Int32 intPotNo)
CTools2	String* GetToolKind(Int32 intPotNo)
CTools2	Boolean IsToolInUse(Int32 intPotNo) void SetToolInUse(Int32 intPotNo, Boolean blnValue)
CTools2	Boolean IsStandardTool(Int32 intPotNo) void SetStandardTool(Int32 intPotNo, Boolean blnValue)
CTools2	Boolean IsAdjustmentTool(Int32 intPotNo) void SetAdjustmentTool(Int32 intPotNo, Boolean blnValue)
CTools2	CarrierStatusEnum GetCarrierStatus(Int32 intPotNo) ;
CTools2	ToolLifeModeEnum GetMode(Int32 intPotNo) void SetMode(Int32 intPotNo, ToolLifeModeEnum enValue)
CTools2	ToolLifeStatusEnum GetStatus(Int32 intPotNo)
CTools2	Int32 GetToolLife(Int32 intPotNo) void SetToolLife(Int32 intPotNo, Int32 intValue)
CTools2	Int32 GetToolLifeRemaining(Int32 intPotNo) void SetToolLifeRemaining(Int32 intPotNo, Int32 intValue)
CTools2	Int32 GetToolLifeRemainingTimeSecond(Int32 intPotNo)
CTools2	Double GetToolLengthOffset1(Int32 intPotNo) ArrayList* GetToolLengthOffset1(Int32 intFromPotIndex, Int32 intToPotIndex) void SetToolLengthOffset1(Int32 intPotNo, Double dblValue) void AddToolLengthOffset1(Int32 intPotNo, Double dblValue) void CalToolLengthOffset1(Int32 intPotNo, Double dblValue)
CTools2	Double GetToolLengthOffset2(Int32 intPotNo) ArrayList* GetToolLengthOffset2(Int32 intFromPotIndex, Int32 intToPotIndex) void SetToolLengthOffset2(Int32 intPotNo, Double dblValue) void AddToolLengthOffset2(Int32 intPotNo, Double dblValue) void CalToolLengthOffset2(Int32 intPotNo, Double dblValue)
CTools2	Double GetToolLengthOffset3(Int32 intPotNo) ArrayList* GetToolLengthOffset3(Int32 intFromPotIndex, Int32 intToPotIndex) void SetToolLengthOffset3(Int32 intPotNo, Double dblValue) void AddToolLengthOffset3(Int32 intPotNo, Double dblValue) void CalToolLengthOffset3(Int32 intPotNo, Double dblValue)
CTools2	Double GetCutterRCompOffset1(Int32 intPotNo) void SetCutterRCompOffset1(Int32 intPotNo, Double dblValue) void AddCutterRCompOffset1(Int32 intPotNo, Double dblValue) ArrayList* GetCutterRCompOffset1(Int32 intFromPotIndex, Int32 intToPotIndex)

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

CTools2	Double GetCutterRCompOffset2(Int32 intPotNo) void SetCutterRCompOffset2(Int32 intPotNo, Double dblValue) void AddCutterRCompOffset2(Int32 intPotNo, Double dblValue) ArrayList* GetCutterRCompOffset2(Int32 intFromPotIndex, Int32 intToPotIndex)
CTools2	Double GetCutterRCompOffset3(Int32 intPotNo) void SetCutterRCompOffset4(Int32 intPotNo, Double dblValue) void AddCutterRCompOffset3(Int32 intPotNo, Double dblValue) ArrayList* GetCutterRCompOffset3(Int32 intFromPotIndex, Int32 intToPotIndex)
CTools2	Double GetToolLengthWearOffset(Int32 intPotNo) void SetToolLengthWearOffset(Int32 intPotNo, Double dblValue) void AddToolLengthWearOffset(Int32 intPotNo, Double dblValue) ArrayList* GetToolLengthWearOffset(Int32 intFromPotIndex, Int32 intToPotIndex)
CTools2	Double GetCutterRCompWearOffset(Int32 intPotNo) void SetCutterRCompWearOffset(Int32 intPotNo, Double dblValue) void AddCutterRCompWearOffset(Int32 intPotNo, Double dblValue) ArrayList* GetCutterRCompWearOffset(Int32 intFromPotIndex, Int32 intToPotIndex)
CTools2	ToolTypeEnum GetToolType(Int32 intPotNo) void SetToolType(Int32 intPotNo, ToolTypeEnum enValue) ArrayList* GetToolType(Int32 intFromPotIndex, Int32 intToPotIndex)
CTools2	Double GetToolAngle(Int32 intPotNo) void SetToolAngle(Int32 intPotNo, Double dblValue) void AddToolAngle(Int32 intPotNo, Double dblValue) ArrayList* GetToolAngle(Int32 intFromPotIndex, Int32 intToPotIndex)
CTools2	Double GetToolDiameter(Int32 intPotNo) void SetToolDiameter(Int32 intPotNo, Double dblValue) void AddToolDiameter(Int32 intPotNo, Double dblValue) ArrayList* GetToolDiameter(Int32 intFromPotIndex, Int32 intToPotIndex)
CTools2	Double GetToolNoseDiameter(Int32 intPotNo) void SetToolNoseDiameter(Int32 intPotNo, Double dblValue) void AddToolNoseDiameter(Int32 intPotNo, Double dblValue) ArrayList* GetToolNoseDiameter(Int32 intFromPotIndex, Int32 intToPotIndex)

The following functions of Okuma.CMCMDAPI.dll library will be only available based on the version of NC control software:

Class	Interfaces
CATC2	Sub UnRegisterToolPot(ByVal intPotNo As Integer, Optional ByVal intToolNo As Integer = 0, Optional ByVal blnRandomATC As Boolean = False)



CATC2	RegisterToolPot(ByVal intToolNo As Integer, ByVal intPotNo As Integer, ByVal intGroupNo As Integer, ByVal intSerialNo As Integer, Optional ByVal blnDummyTool As Boolean = False, Optional ByVal strToolName As String = "")
CATC2	RegisterToolPot(ByVal intPotNo As Integer, ByVal intGroupNo As Integer, ByVal intSerialNo As Integer, Optional ByVal blnDummyTool As Boolean = False, Optional ByVal strToolName As String = "")
CATC2	SetToolCarrierStatus(ByVal intPotNo As Integer, ByVal enCarrierStatus As CarrierStatusEnum)
CATC2	SetToolKind(ByVal intPotNo As Integer, ByVal enToolKind As ToolKindEnum)
CATC2	SetToolLifeStatus(ByVal intPotNo As Integer, ByVal enToolLifeStatus As ToolLifeStatusEnum)

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

Classes	Interfaces
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) ;
CTools2	All functions in CTools2 class

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

Classes	Interfaces
CATC2	All functions in CATC2 class
CMachine	SetUserAlarm(ByVal enAlarm As UserAlarmEnum, Optional ByVal strAlarmMessage As String = "")
CMachine	ClearUserAlarmD()

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

For questions about these functions, please consult with your Okuma representatives.

### 3. Features

Please refer the help document for more information of new/update functions.

#### 3.1 DATA-API

STATE	Control	Classes	Interfaces
NEW	P300	CATC	GetATCPanelErrorCode ()
NEW	P300	CATC	GetATCPanelPotNo()
NEW	P300	CATC	GetATCPanelToolNo()
NEW	ALL	CATC	ATCSubPanelOperationModeEnum GetATCPanelOperationMode() ; void SetATCPanelOperationMode(ATCSubPanelOperationModeEnum enValue) ;
NEW	P300	CTools	GetToolList(ToolListTypeEnum enToolListType)
UPDATE	P200 P300	CIO	OnOffStateEnum GetUserTaskIOVariable(IOTypeEnum enIO, Int32 intIndex) ;
UPDATE	P200 P300	CIO	OnOffStateEnum GetUserTaskIOVariable(IOTypeEnum enIO, Int32 intIndex) ; void SetUserTaskOutputVariable(Int32 intIndex, OnOffStateEnum enValue) ;
UPDATE	P200 P300	CIO	OnOffStateEnum GetProtectedUserTaskOutputVariable(Int32 intIndex) ;
UPDATE	P200 P300	CIO	void SetProtectedUserTaskOutputVariable(Int32 intIndex, OnOffStateEnum enValue) ;

#### 3.2 Command API

STATE	Control	Classes	Interfaces
NEW	P300	CATCPanel	void ClearError()
NEW	P300	CATCPanel	int SearchPotNo(String^ strToolName)
NEW	P300	CATCPanel	int SearchPotNo(int intGroupNo, int intSerialNo)
NEW	P300	CATCPanel	void SetIndexPotNo(int intIndexPotNo)

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

## 4. Known Bugs and Limitations

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

### 4.1 Defect

#### 4.1.1 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.

Solution/Work-arounds: A bat file is created during the installation of THINC-API and will be run 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. All applications should use new version THINC-API if possible to prevent using Firebird database and to be compatible with Windows 10 once it is available.

#### 4.1.2 Data-API

##### 4.1.2.1 CAxis class

Function: AxisTypeEnum GetAxisType (AxisIndexEnum enValue)

Symptom: Failed to access machine data for YS axis. YS axis is available in NC-HMI.

Solution/Work around: None.

All functions having YS axis as input parameter are affected by GetAxisType function and fail to get/set NC data.

Function: String\* GetAxisName (AxisIndexEnum enValue)

Symptom: Failed to access machine data for YS axis. YS axis is available in NC-HMI.

Solution/Work around: None.

##### 4.1.2.2 CMOPTool class

Function: GetToolDataNumber(ByVal *intToolNo* As Integer, ByVal *intClassNo* As Integer) As Integer

Symptom: This function cannot get tool data number given tool number greater than 300.

Solution/Work around: None.

##### 4.1.2.3 CSpindle class

Function: SpindleStateEnum GetSpindleState()

Symptom: Function return incorrect state - When machine is in Manual mode, the spindle is set to ORIENTATION mode.

NC-HMI and THINC-API library returns the same state. Then, machine mode is changed to either AUTO or MDI, NC\_HMI still report the same state as before, but THINC-API library reports spindle in stop mode.

Solution/Work around: None

THINC-API	Version: S5015-003-28
Release Notes For Machining Center	Date: 12/07/2017

## 4.2 Defects Fixed in this Release

None