

• 특집 • 공작기계의 스마트화 기술

# 공작기계 상태 모델 기반의 이기종 CNC 데이터 통합 인터페이스 개발

## Development of Unified Interface for Multi-Vendors' CNC Based on Machine State Model

윤주성<sup>1,#</sup>, 박일하<sup>1</sup>, 손진호<sup>2</sup>, 김헌정<sup>3</sup>

Joo Sung Yoon<sup>1,#</sup>, Il Ha Park<sup>1</sup>, Jin Ho Sohn<sup>2</sup>, and Hoen Jeong Kim<sup>3</sup>

<sup>1</sup> 한국생산기술연구원 스마트제조기술그룹 (Smart Manufacturing Technology Group, Korea Institute of Industrial Technology)

<sup>2</sup> 현대위아 소프트웨어개발팀 (S/W Development Team, HYUNDAI WIA Co., Ltd.)

<sup>3</sup> 씨에스캠 소프트웨어사업부 (S/W Business Department, CSCAM Co., Ltd.)

# Corresponding Author / E-mail: jsyoon@kitech.re.kr, TEL: +82-41-589-8408

KEYWORDS: Machine tool HMI (공작기계용 HMI), Machine tool data schema (공작기계 데이터 구조), Machine state model (공작기계 상태 모델), Multi-vendors' CNCs (이기종 CNC), Unified CNC interface (통합CNC 인터페이스)

*The Human Machine Interface (HMI) is dependent on the Computerized Numerical Control (CNC). As a result, the HMI software is developed according to the interface method of each CNC in duplicate. Even though the same function of HMI software module is developed using the same data item, it must be implemented separately in different development environments. This is because each CNC has a different address system and provides unique data interface method. In this study, we proposed a unified interface system that can standardize and integrate data interface method to support the development of HMI software module applicable to machine tools adopting multi-vendors' CNCs. To clearly define this work, we developed new parameters, methods, and address system based on the machine state model, composed of data elements of machine tool structure, process, and status, provides the same interface by capsulizing existing CNC interfaces. The proposed interface system is designed to provide an API (Application Programming Interface) in the form of a library. The implementation architecture is designed and details of the operating logic within the detailed components and interfaces are elaborated. The implementation and test results are illustrated to verify an application example of the proposed unified interface system.*

Manuscript received: December 11, 2017 / Revised: January 24, 2018 / Accepted: January 25, 2018

### 1. 서론

ICT (Information and Communication Technology) 기술의 발전과 제조-ICT 융합 추세가 가속화됨에 따라 가공시스템의 경쟁력이 정밀도, 고속화, 내구도 등 기계 하드웨어 중심에서 환경인지, 장비 간 연결을 통한 기계장비 스마트화 기술 경쟁으로 빠르게 전환되고 있다. 특히 제조현장에서 요구하는 복잡한 계산 및 빠른 통신 속도를 CNC 내에서 제공받는 것이 어려워지면서 공작기계 장비에 최적화된 고성능 HMI에 대한 요구가 증대되고, 애플리케이션 형태로 확장 가능한 HMI 소프트웨어 모듈이 탑재된

공작기계가 개발되고 있는 추세이다.<sup>1-3</sup> 기존의 CNC 시스템에서 제공하는 기본 조작화면과 제한된 입력방식을 사용하는 전통적인 HMI 형태에서 스마트폰과 같은 멀티 터치 인터페이스를 채용하여 사용 편의성을 높이고 공작기계 특성을 반영한 다양한 기능들을 애플리케이션 단위로 탑재하여 기능 확장성을 제고하는 형태로 진화하고 있는 것이다.

이러한 새로운 형태의 고성능 HMI가 공작기계 장비의 핵심요소로 부각되면서 CNC에 종속적인 HMI 소프트웨어 모듈 개발의 비효율성 문제가 중요한 이슈이다. 고객의 주문에 따라 서로 다른 기종의 CNC를 채용하여 장비를 제작할 때 같은 기능의 HMI 소

소프트웨어 모듈을 CNC에서 제공하는 주소체계와 인터페이스 방식에 따라 중복적으로 개발해야 하는 문제가 발생할 수 있다. CNC마다 서로 다른 형태의 주소체계를 사용할 뿐 아니라 각자 고유의 데이터 인터페이스 방식을 제공하기 때문에 동일한 데이터 항목으로 동일한 기능을 구현하더라도 서로 다른 개발 환경에서 개별적으로 구현해야 하는 것이다. 실제로 공작기계 개발 업체들에서는 장비에 탑재할 CNC의 기종 별로 CNC 업체에서 제공하는 HMI 개발환경에 따라 추가적인 소프트웨어 모듈을 개발하고 있으며, 개발 인력이나 조직도 CNC 기종에 따라 나뉘게 되어 CNC에 대한 종속성이 커지게 된다.

이러한 HMI 소프트웨어 개발의 중복성 문제를 해결하고 HMI 소프트웨어 모듈의 재활용성을 높이기 위해, 본 연구에서는 이기종 CNC를 채용하는 공작기계용 HMI 소프트웨어 개발에 공통 적용 가능한 통합 인터페이스를 라이브러리 형태로 개발하고 이를 API로 제공하고자 한다. 통합된 HMI와 관련하여, CNC 기종에 관계없이 동일한 HMI를 채용한 공작기계가 개발되고 있으나<sup>3</sup> 이는 디자인과 UX (User Experience)/UI (User Interface)를 동일하게 맞추는 것에 집중하고 있다. 또한 공작기계의 이상상태를 파악하며 제어하고 관련 데이터를 분석할 수 있는 HMI 시스템에 대한 연구가 있으나<sup>4</sup> 센서 인터페이스와 복수장비 및 주변장치와의 통합을 다루고 있으며, 하나의 HMI를 통해 생산현장의 다양한 설비의 상태 모니터링에 대한 연구가 있으나,<sup>5</sup> 다양한 설비에 특화된 HMI 화면 재구성에 초점을 맞추고 있다. 인터페이스 표준화와 관련해서는 MTConnect와 같은 장비 인터페이스 표준이 제공되고 있으나<sup>6</sup> 이는 기종에 따라 인터페이스 모듈을 각기 개발해야 하며 네트워크 연결용으로서 성능적으로 실시간 데이터를 다루는 HMI 소프트웨어 개발용으로 사용하기에는 한계가 있다. MTConnect 기반으로 이종 기기들에 대한 생산장비 오류 검출 및 상태 관리 시스템을 제시한 연구나<sup>7</sup> 에너지 분석 및 예측 모델링 기법에 관한 연구가 있으나,<sup>8</sup> 운영관리 수준의 시스템 활용을 위한 연구들로 HMI 개발을 다루지는 않는다. 이기종 CNC 공작기계에 공통으로 적용할 수 있는 HMI 소프트웨어 모듈 개발을 위한 통합 인터페이스를 다루는 연구는 미흡한 실정이다.

본 논문은 다음과 같이 구성된다. 2장에서는 이기종 CNC에 공통 적용 가능한 HMI 시스템의 통신 구조를 도출하기 위한 공작기계 상태 모델을 제시하고, 3장에서는 이를 기반으로 개발된 이기종 CNC 통합 인터페이스의 상세 모듈 구성과 구동 로직을 구체화한다. 4장에서는 개발된 통합 인터페이스 라이브러리와 이를 활용한 테스트 App을 통한 작동성 테스트 결과를 제시하여 제안된 통합 인터페이스의 적용 예시를 설명한다.

## 2. 공작기계 상태 모델 기반 HMI 통신 구조

### 2.1 이기종 CNC 통합 HMI 통신 구조

본 연구에서는 이기종 CNC에 대한 HMI를 개발할 때에 동일한 기능의 소프트웨어 모듈은 서로 다른 CNC를 채용하는 공작기

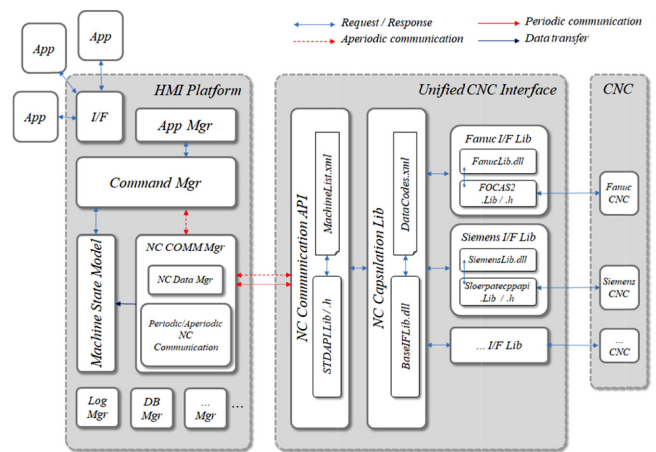


Fig. 1 Architecture for communication between HMI and multi-vendor CNCs

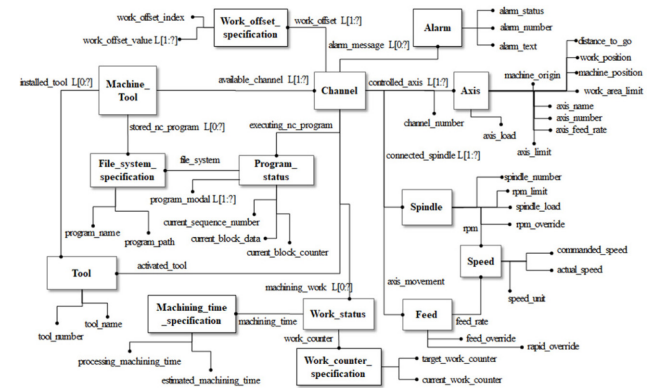


Fig. 2 Data schema for machine state model

계에서도 공통으로 사용할 수 있도록 통합된 인터페이스 방식을 제시한다. 구체적으로는 Fig. 1과 같이 공작기계의 구성, 상태, 공정 등에 대한 데이터 요소를 구조화한 공작기계 상태 모델을 생성하고, CNC에서 공작기계 상태 모델(Machine State Model)에 주기적으로 실시간 데이터를 업데이트할 수 있도록 연결하여, HMI 소프트웨어 모듈(App)에서는 직접 CNC에 접속하지 않고 공작기계 상태 모델에서 실시간 데이터를 읽어 온다. 여기에서 CNC로부터 공작기계 상태 모델을 연결하는 이기종 CNC 통합 인터페이스(Unified CNC Interface)는 이기종 CNC 고유의 인터페이스 방식을 표준화하여 캡슐화된 형태의 새로운 인터페이스 메소드를 정의하고 공작기계 상태 모델의 데이터 요소에 동기화시키는 방식으로 작동한다.

### 2.2 공작기계 상태 모델 데이터 구조

본 연구에서 제안하는 공작기계 상태 모델은 공작기계의 구조, 공정, 상태 등에 대한 데이터 요소를 구조화하여 가상의 소프트웨어 시스템 상에 올릴 수 있도록 함으로써, HMI가 CNC 기종에 관계없이 데이터 통신할 수 있도록 지원하는 핵심 컴포넌트이다. 공작기계 상태 모델의 데이터 구조는 Fig. 2와 같으며, CNC에 중립적인 데이터 요소 추출을 위해 ISO 14649 표준의 공정, 공구,

장비 데이터 모델과 상용 CNC들의 주소체계 및 데이터 인터페이스 API를 분석한 결과를 토대로 표준화하여 EXPRESS-G 다이어그램으로 도식화한 것이다.

전체적인 데이터 구조는 하나의 공작기계가 여러 개의 채널을 가질 수 있고, 각 채널은 여러 개의 축과 하나 혹은 그 이상의 스핀들로 구성되며 이들의 상대 움직임에 따른 이송 운동, 장착된 공구, 실행 중인 NC 프로그램, 가공공정 진행 상황, 이상에 따른 알람 발생 등의 다양한 상태정보를 나타낼 수 있다. 또한 채널과 별개로 공작기계 내에 존재하는 공구 리스트와 메모리에 저장된 프로그램 파일 리스트가 존재한다. 단위 공작기계를 식별하는 Machine\_Tool 엔티티는 기본적인 장비 구성 정보와 함께 가용 채널, 보유 프로그램 리스트, 보유 공구 리스트 등의 정보를 포함하며, 대부분의 상태 데이터가 각 채널에 따라 상세 속성으로 정의된다.

채널 별로 기록되는 상태 데이터는 Channel 엔티티로 정의되며, 이는 채널의 식별자 정보와 함께 제어 가능한 축 정보, 연결된 스핀들 정보, 이송 상태 정보, 작업 상태 정보, 활성화된 공구 정보, 실행 중인 프로그램 정보를 포함한다. 제어 가능한 축에 대한 데이터들은 Axis 엔티티로 정의되며, 이는 해당 축 식별 정보와 함께 축별로 기록되는 위치 정보, 부하 정보 등과 같은 상태정보를 포함한다. 스핀들별로 기록되는 상태정보를 표현하는 Spindle 엔티티는 해당 스핀들의 식별 정보, 속도 정보, 속도에 대한 오버라이드, 부하 정보, 한계 속도 등으로 구체화될 수 있고, 축 이송에 대한 속도 관련 정보를 표현하는 Feed 엔티티는 가공 이송속도, 가공 이송속도에 대한 오버라이드, 급속 이송에 대한 오버라이드 정보 등을 포함한다.

Tool 엔티티는 공구에 대한 식별자, 기하학적 사양 등을 포함하며 현재 가공 중인 절삭공구와 NC에 등록된 절삭공구 리스트에 대한 데이터 타입으로 사용될 수 있다. 또한 File\_System\_Specification 엔티티는 NC 프로그램 파일에 대한 파일명, 저장 경로 등의 기본정보로 정의되어 메모리에 저장된 NC 프로그램 파일 리스트에 대한 데이터 타입으로 사용될 수 있으며, 실행 중인 NC 프로그램의 진행 상태를 정의하는 Program\_Status 엔티티는 File\_System 타입의 프로그램 기본 정보와 함께 프로그램 내의 현재 시퀀스 정보, 블록 카운터 정보 등을 포함한다.

그 외에 Work\_Status 엔티티는 가공 수량 정보, 가공 시간 등 장비에 할당된 가공 작업들의 진척 상태를 나타내며, Alarm 엔티티는 해당 채널에서 발생한 알람에 대한 정보를 나타내며 알람 유형, 알람 번호, 알람 발생/해제 시간, 알람 메시지 등을 포함한다. 또한 Work\_Offset\_Specification 엔티티는 채널 내 공작물 좌표계 관련 G 코드에 대한 정보를 나타내며, G 코드 인덱스, 축 별 오프셋 량 리스트 등으로 구체화될 수 있다.

### 3. 이기종 CNC 통합 인터페이스 작동 방식

#### 3.1 인터페이스 상세 모듈 구성

2장에서 정의한 공작기계 상태 모델을 레퍼런스 스키마로 참조

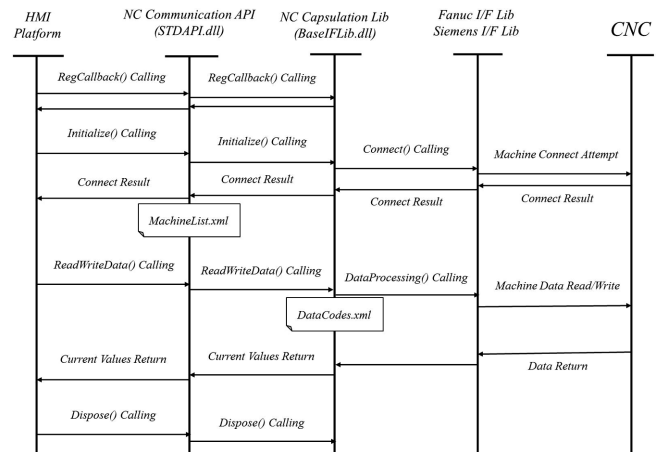


Fig. 3 Operating sequence of unified CNC interface

하여 Fanuc과 Siemens 제어기에 대한 이기종 CNC 통합 인터페이스를 구현하였으며, Fig 3과 같이 (1) NC Communication API, (2) NC Capsulation Library, (3) Fanuc, Siemens Interface Library로 구성된다.

NC Communication API는 HMI 플랫폼에서 호출 가능한 함수들을 제공하는 모듈로서, 데이터 요청이 오면 연결된 공작기계의 구성정보를 담고 있는 MachineList.xml 파일을 읽어 연결 성공한 CNC의 라이브러리 파일을 로드한다. 뿐만 아니라 API 초기화 및 해제, IP 주소 유효성 검사 등과 같이 통합 인터페이스가 HMI 플랫폼 혹은 이기종 제어기와 통신할 때 기능을 안정적으로 수행할 수 있도록 지원하는 함수들을 제공한다.

NC Capsulation Library는 서로 다른 인터페이스 방식을 사용하는 CNC를 표준화하여 새로운 파라미터, 메소드, 주소체계를 정의하고 기존의 CNC 인터페이스를 감싸서 동일한 인터페이스를 제공할 수 있도록 관리한다. 이 때 새롭게 정의되는 파라미터, 메소드, 주소체계는 2장에서 제시한 공작기계 상태 모델 데이터 구조에 따라 최하위 데이터 속성에 대한 데이터 코드를 부여하고 해당 데이터를 읽고 쓸 수 있는 표준 함수를 생성하는 방식으로 개발되었으며, 이에 대한 매핑 관계는 DataCodes.xml에 정의하고, 요청되는 데이터 코드에 따라 DataCodes.xml을 읽어 해당되는 표준 함수를 호출하고 실행시킨다.

Fanuc, Siemens Interface Library는 Fanuc과Siemens 에서 제공하는 API를 캡슐화하여 제공한다. 표준 함수에 대한 실제 구현문이 위치하며, 함수 내부에는 각 CNC에서 제공하는 함수 혹은 어드레스 정보를 포함하여 고유의 인터페이스 방식에 따라 CNC에 접근하게 된다.

세부 작동 시퀀스는 다음과 같다: HMI 플랫폼에서 데이터 호출 시 RegCallback( )를 통해 Callback Method를 등록하고, Initialize( )를 통해NC Communication API를 초기화한다. Connect( )를 사용하여 제어기에 연결을 시도한 후 결과를 플랫폼에 반환하고, 플랫폼에서는 ReadWriteData( )를 통해 원하는 데이터 코드를 호출한다. 데이터 코드와 매핑된 표준 함수를 실행하여 Ethernet 통신을 통해 읽어 온 데이터 결과를 반환하며, 프로그램 종료 시

Table 1 Example of data codes for position data

Target data	Data code	
	Number	String
machinePosition	21110	machine_channel_axis_machinePosition
workPosition	21111	machine_channel_axis_workPosition
distanceToGo	21112	machine_channel_axis_distanceToGo
relativePosition	21113	machine_channel_axis_relativePosition

Dispose( )를 통해 리소스를 해제한다.

### 3.2 인터페이스 구동 로직

개발된 통합 인터페이스는 HMI 소프트웨어 모듈에서 CNC 데이터를 읽거나 쓰는 기능을 모두 지원하며, HMI 플랫폼의 통신 관리자(Communication Manager)가 공작기계 상태 모델을 통해 HMI 소프트웨어 모듈과 CNC 간의 통신을 증계한다. 이하로는 데이터 읽기 요청에 대한 인터페이스 구동 로직에 대해 상세히 기술한다.

HMI 소프트웨어 모듈에서는 URI 주소체계 형태로 데이터 요청을 실행하면 해당 주소는 HMI 플랫폼을 통해 공작기계 상태 모델에서 정의된 데이터 항목에 매핑된 데이터 코드로 변환된다. 데이터 코드는 HMI 플랫폼과 통합 인터페이스가 커뮤니케이션 할 수 있도록 공작기계 상태 모델에서 정의된 데이터 항목을 기준으로 데이터 구조 및 의미를 파악할 수 있는 문자열과 5자리 정수로 정의된다. Table 1은 위치 데이터에 대한 데이터 코드 예시를 나타낸다.

HMI 플랫폼으로부터 데이터 요청을 전달 받을 수 있도록 통합 인터페이스에서는 ReadWriteData( )를 API로 제공하고 Fig. 4의 구조체 형태의 파라미터를 사용하여 데이터를 요청하고 값을 반환한다. 이 구조체를 통해서 전달되는 정보는 인터페이스 하고자 하는 데이터 항목에 대응되는 데이터 코드(DEFINE\_NUMBER), 읽기와 쓰기 요청을 구별하기 위한 구분자(RW)와 해당 데이터에 접근하기 위해 필요한 인덱스 번호(i, j, k)를 포함한다. 인덱스 번호 세 가지는 고정된 값이 아니며 데이터 항목마다 다른 종류를 가진다. 인덱스 번호의 종류는 상태 모델 데이터 구조 내에서 데이터 항목 별 상위 엔티티에 따라 결정된다. 예를 들어 Channel 엔티티와 Axis 엔티티 하위 데이터 항목들은 i=채널 번호, j=축 번호의 인덱스를 필요로 한다. 요청에 대한 반환 값(value)은 문자열, 정수형, 실수형 단일 값과 배열 값을 지원하며, 요청의 성공 여부(STATUS)를 확인할 수 있도록 구조체 안에 저장된다.

통합 인터페이스에서는 구조체 형태의 데이터 요청을 받아 Fig. 5의 예시와 같이 DataCodes.xml 파일에 선언된 메소드 매핑 관계에 따라 표준함수를 호출하게 된다. DataCodes.xml 파일 내에는 코드체계 규칙에 따른 데이터 코드, 제어기 유형, 읽기 요청 시 사용할 Get 함수와 쓰기 요청 시 사용할 Set 함수, 다중 읽기

```
typedef struct cncdata
{
    int DEFINE_NUMBER;
    char DESC[MAX_DESC_STRING_SIZE];
    int i;
    int j;
    int k;
    DATASTATUS STATUS;
    DIRECTION RW;
    RETURNVALUE value;
} CNCDATA;
```

Fig. 4 Data type for query and return between HMI and unified CNC interface

```
<DataCode DefineNumber="21110" Desc="machine.channel.axis.machinePosition">
<MethodInfo Type="Fanuc" ReadMethod="getMachineAxisPosition" WriteMethod="" DataAddress="getAxisData" StartIndex="0" />
<MethodInfo Type="SIEMENS" ReadMethod="getMachineAxisPosition" WriteMethod="" DataAddress="/Channel/MachineAxis/actToolBasePos[u%d,%d]" StartIndex="1" />
</DataCode>
<DataCode DefineNumber="21111" Desc="machine.channel.axis.workPosition">
<MethodInfo Type="Fanuc" ReadMethod="getAbsoluteAxisPosition" WriteMethod="" DataAddress="getAxisData" StartIndex="1" />
<MethodInfo Type="SIEMENS" ReadMethod="getAbsoluteAxisPosition" WriteMethod="" DataAddress="/Channel/GeometricAxis/actProgPos[u%d,%d]" StartIndex="1" />
</DataCode>
```

Fig. 5 Example of DataCode.xml



Fig. 6 Testbed configuration

또는 쓰기 요청 시 사용할 각 제어기의 파라미터 번호 및 어드레스가 저장되어 있으며, 이러한 메타 데이터에 해당하는 제어기 고유의 인터페이스 메소드를 호출하여 CNC에서 데이터를 읽거나 쓸 수 있도록 지원한다.

## 4. 상태 모델 및 통합 인터페이스 구현 테스트

### 4.1 개발 및 테스트 환경 구축

이기중 CNC 통합 인터페이스에 대한 개발 및 테스트 환경은 Fig. 6와 같이 Fanuc, Siemens 제어기의 NC커널, X/Y/Z축과 스핀들에 대한 모터 드라이브 모듈, HMI로 구성된 시뮬레이터, 시뮬레이터와 Ethernet 통신으로 연결된 워크스테이션으로 구성된다. 제안된 공작기계 상태 모델과 이기중 CNC 통합 인터페이스에 대한 검증 테스트를 위해 산업용으로 가장 많이 사용되는 제어

Table 2 Example of standard functions for position data

Target data	Standard function	Siemens address	Fanuc API
machinePosition	getMachineAxisPosition	/Channel/MachineAxis/ actToolBasePos[u%d,%d]	cnc_machine(unsigned short FlibHndl, short axis, short length, ODBAXIS *machine)
workPosition	getAbsoluteAxisPosition	/Channel/GeometricAxis/ actProgPos[u%d,%d]	cnc_absolute(unsigned short FlibHndl, short axis, short length, ODBAXIS *absolute)
distanceToGo	getDistanceToGoPosition	/Channel/MachineAxis/ toolBaseDistToGo[u%d,%d]	cnc_distance(unsigned short FlibHndl, short axis, short length, ODBAXIS *distance)
relativePosition	getRelativePosition	/Channel/GeometricAxis/ aaAcsRel[u%d, %d]	cnc_relative(unsigned short FlibHndl, short axis, short length, ODBAXIS *relative)

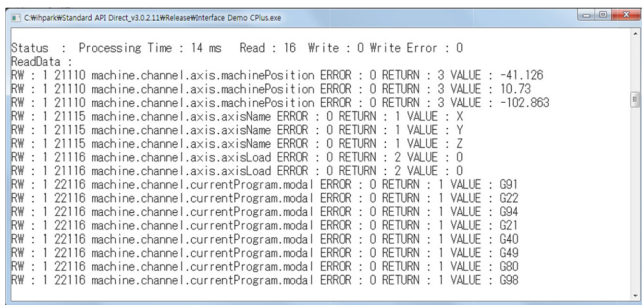


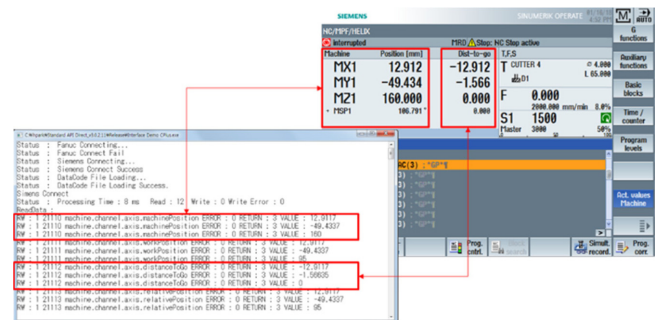
Fig. 7 Example of test app command

기들 중에 주소체계와 인터페이스 방식이 다른 Siemens, Fanuc 제 어기를 대상으로 선정하고, Siemens는 840D 모델에 대한 시물레이 터를, Fanuc은 31i-B 모델로 구성하였다. 또한 제안된 공작기 계 상태 모델을 참조하여 139종의 표준함수를 라이브러리로 구현 하고 이를 호출할 수 있는 API를 통해 테스트 App을 개발하였으 며, 워크스테이션에 해당 라이브러리와 테스트 App을 탑재하여 작동성 테스트를 수행하였다. 개발 환경은 .NET Framework 4.5 기반의 Visual Studio 2012 professional에서 C#을 사용하였으며, Siemens의 Programming Package 4.7와 Fanuc의 FOCAS2 Library 4.2를 기본 라이브러리로 채용하였다.

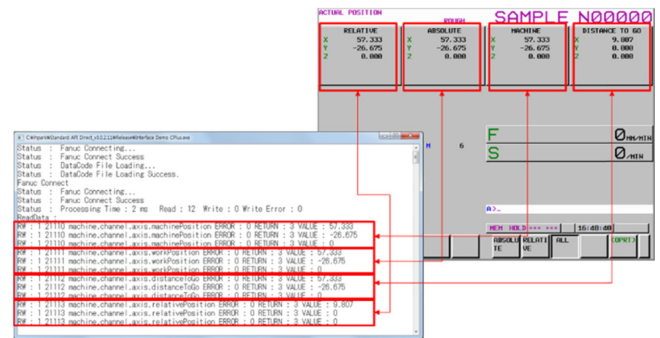
4.2 테스트 결과

통합 인터페이스 작동성 테스트 진행 방식은 다음과 같다. 각 CNC 시물레이터에 동일한 ISO 코드로 구성된 NC 프로그램을 실행시키고, Fig. 7과 같은 테스트 App 커맨드에서 나타나는 데 이터 값과 각 시물레이터의 HMI 화면에 출력되는 값을 비교한다. 쓰기의 경우는 테스트 App 커맨드를 통해 동일한 값을 입력하고 각 시물레이터의 HMI 화면에 해당 값이 제대로 반영되는지 확인 한다. 동일한 테스트 App을 사용하여 각 시물레이터 별로 같은 테스트를 순차적으로 반복 시행한다.

Fig. 8은 테스트 예시 화면이다. 축의 위치와 관련된 4가지 유 형의 데이터 항목, 머신좌표, 절대좌표, 남은 거리, 상대좌표를 읽 어 오기 위해서는, 테스트 App에 API ReadWriteData( )에 구조 체 cncdata 타입 파라미터의 속성에 따라 DEFINE\_NUMBER는 4개 데이터 항목의 데이터 코드, i는 채널 번호, j는 축 번호, RW는 READ를 입력값으로 할당하여 실행하면, 통합 인터페이스에서는 표준함수 getMachineAxisPosition( ), getAbsoluteAxisPosition( ),



(a) Siemens



(b) Fanuc

Fig. 8 Screen shot of test example

getDistanceToGoPosition( ), getRelativePosition( )를 호출하여 해당 값을 출력한다. 이때 연결되어 있는 CNC 유형은 통신 시작과 함께 대상 공작기계에서 받아 와서 MachineList.xml에 저장되어 있으며, MachineList.xml의 CNC 유형 정보와 구조체 cncdata 타입 파라미터로 넘어 온 메타데이터에 따라 DataCode.xml를 통해 해당 표준함수를 호출하게 되는 방식이다. 테스트 App 커맨드에 출력되는 데이터를 CNC 시물레이터의 HMI 화면에 출력된 값과 비교하면서 139종의 표준함수를 테스트하였으며, 해당 데이터 값 이 일치하는 것으로 정상 작동하는 것을 확인하였다. Table 2는 위치 데이터 인터페이스에 대한 표준함수 예시를 나타낸다.

5. 결론

본 연구에서는 이기중 CNC 공용 HMI 소프트웨어 모듈에 대

한 데이터 통합 인터페이스를 제안하고, 공작기계 상태 모델과 이기종 CNC 인터페이스 캡슐라이징 개념을 도입하여 이를 구현하고 작동성을 검증하였다. 공작기계 상태 모델에 대한 데이터 구조를 정의하고, 이를 레퍼런스 스키마로 참조하여 서로 다른 주소체계와 인터페이스 방식을 캡슐화하여 표준함수를 작동시키는 통신 구조를 구체화함으로써 통합 인터페이스를 개발하였다.

본 연구를 통해 개발된 통합 인터페이스는 이기종의 CNC를 채용하여 공작기계에 탑재될 HMI 소프트웨어를 개발할 때, 특정 CNC의 인터페이스 방식을 따로 파악할 필요 없이 공작기계 상태 모델을 통해 일원화된 방식으로 인터페이스를 제공할 수 있으며, 동일한 소프트웨어 모듈은 중복 개발 필요 없이 공통 적용 가능하여 CNC에 종속된 개발로 인한 비효율성을 제거할 수 있다. 뿐만 아니라 공작기계 사용자 관점에서도 CNC에 따라 다른 조작 화면을 사용할 필요 없이 CNC 기종이 어떤 것인지 알지 못해도 단일한 방식으로 기계를 조작할 수 있는 편의성을 제공한다.

본 연구에서 제시한 공작기계 상태 모델 및 이종 CNC 통합 인터페이스는 현재 2종의 이기종 CNC에 대한 139종의 표준함수를 제공하고 있으며, 다양한 HMI 소프트웨어 모듈 개발에 사용하여 테스트 결과를 피드백함으로써 추가적인 데이터 항목을 업데이트하고 있다. 추후에 대상 제어를 3종으로 확대하고, 추가적인 상용 CNC 분석을 통해 새로운 데이터 항목을 반영함으로써 상태 모델을 확장하는 것을 계획하고 있다. 뿐만 아니라 상용화를 고려하여 HMI 소프트웨어 적용 유즈케이스를 늘려 가면서 반복적인 테스트를 수행함으로써 안정성을 확보해야 할 것이다.

## ACKNOWLEDGEMENT

본 연구는 산업통상자원부의 산업핵심기술개발사업의 지원으로 수행되었음[과제명: 공작기계용 지능형 HMI 플랫폼 및 핵심 운용모듈 개발(과제번호: 10067410)].

## REFERENCES

1. Okuma, <https://www.okuma.co.jp/english/onlyone/osp-suite/> (Accessed 22 JAN 2018)
2. Mazak, <https://www.mazakusa.com/machines/technology/smooth-technology/> (Accessed 22 JAN 2018)
3. DMG-Mori, <https://celos.dmgmori.com/> (Accessed 22 JAN 2018)
4. Choi, D. B., Kim, S. H., Lee, D. K., and Jeong, S. I., "HMI Configuration of ICT Machine Tools with Built-in Sensor," Proc. of the Korean Society of Manufacturing Technology Engineers Spring Conference, p. 71, 2017.
5. Lee, S. W., Nam, S. J., and Lee, J. K., "Human-Machine Interface Implementation for Diverse Equipments Monitoring," Proc. of the Korean Society of Manufacturing Technology Engineers Autumn Conference, p. 9, 2014.
6. The Association for Manufacturing Technology, "MTConnect® Standard Part 1 - Overview and Protocol Version 1.3.0," Standard-MTConnect, 2015.
7. Lee, Y. W., Choi, Y. J., Lee, J. H., Kim, B. G., and Lee, S. W., et al., "Development of Distributed Smart Data Monitoring System for Heterogeneous Manufacturing Machines Operation," Journal of Digital Contents Society, Vol. 18, No. 6, pp. 1175-1182, 2017
8. Shin, S. J., Woo, J. G., Seo, W. C., and Jeong, Y. J., "Energy Prediction Modeling for Numerical Control Programs Using MTConnect," J. Korean Soc. Precis. Eng., Vol. 34, No. 5, pp. 355-362, 2017.