

원전 증기발생기 세관세정용 이동로봇의
3차원 Visualization 및 제어기 개발에 관한 연구

A Study on Development of Controller & 3-D Visualization for
Sludge Lancing of Nuclear Power Steam Generator

김준홍, 조병학

한전전력연구원
대전광역시 유성구 문지동

요 약

원전 증기발생기내의 세관에 점착되는 슬러지를 제거하기 위한 로봇작업을 3차원적으로 표현하는 가시화(Visualization)툴과 이를 이용한 로봇제어기를 개발하였다. 본 툴은 MS사의 Visualization 언어인 DirectX를 이용하여 제작되었으며, 사용된 그래픽모델은 3ds max를 이용하여 제작하였다. 로봇제어기는 원자력환경의 특성상 2개로 분리하여 최소한의 장비만 방사능구역에 위치시키고 나머지 장비는 외부에 위치시켜서 작업구역의 효율적 활용 및 작업 후 제염해체작업의 간략화를 이루었다. 제어프로그램은 모듈형식으로 제작되어 다른 프로그램과 연계하여 제어기를 구성할 수 있으며 다른 분야로의 적용이 매우 용이한 장점이 있다.

Abstract

A visualization tool for representing robot task of sludge lancing in 3-D, and control system for sludge lancing robot is developed. This tool is developed utilizing MS DirectX, and graphic models for this tool is drawn through 3ds max 3.0. On robot controller, we split it in two part. We set the minimum part of controller into CV, and the remaining part out of CV. So we can use the work space efficiently and simplify the decontamination process. Because this tool is developed as module type, One can develop a controller set with another program, and can easily apply it to the other field.

1. 서 론

원자력발전소 증기발생기 내부에는 수천 개의 섬세한 세관이 배열되어 있으며 운전 중 세관에 슬러지가 생성되어 열효율이 떨어지므로 일년에 한차례 있는 정기 보수기간 중 세정로봇을 이용하여 슬러지를 제거하는 작업이 수행된다. 일반적으로 외부의 작업자가 로봇에 부착되어 있는 카메라를 이용하여 작업위치를 파악하여 세관사이에 고압수를 분사하여 작업이 이루어지는데 이 때

작업공간 내부의 고압수, 잔류 증기와 슬러지 등으로 인해 영상의 화질이 떨어져서 작업에 지장을 초래할 수 있으며 제한된 작업공간과 시스템자원의 부족으로 인해 오직 한 대의 카메라로 작업을 하게 되므로 작업자가 관찰할 수 있는 공간도 지극히 제한된다는 문제점이 있다. 또한 고압수를 사용하기 때문에 정확한 지점이 아니라면 증기발생기에 커다란 손상을 초래할 수 있다. 이러한 점을 개선하기 위해 작업공간을 3차원 그래픽으로 모델링하고 카메라 오브젝트를 별도로 지원하는 특징을 갖는 DirectX를 이용하여 작업과정의 가시화를 구현하고 이를 제어에 적용함으로써 다수의 가상카메라로 모든 작업공간을 확인할 수 있고 정확한 작업위치 파악을 가능하게 하여 보다 정확하고 효율적인 작업이 가능하게 하였다. 본 논문은 가시화툴의 제작과 이를 이용한 증기발생기 세정로봇 제어기의 개발에 관하여 다루고 있다.

2. 가시화의 구현

본 시스템 개발에서는 DirectX를 사용하여 3차원 가시화를 구현하였다. DirectX에서는 4×4 행렬을 사용하여 이러한 작업을 수행한다. 본 논문에서는 카메라에 대한 Callback 함수를 작성하고 그 내부에 4×4 행렬을 파라미터로 받아서 프레임의 좌표변환을 처리하는 AddTransform 함수를 사용하였다. 즉 카메라의 위치와 방위를 변환시키면 카메라의 방위각(azimuth) θ , 극각(polar angle) ϕ , 주거리 f 값을 변환시키게 되는데, 이러한 값들이 주어지면 좌표변환 함수의 입력 값으로 사용되는 horizontal 벡터 \bar{H} , vertical 벡터 \bar{V} , approach 벡터 \bar{A} , position 벡터 \bar{C} 를 계산할 수 있다. 다음은 카메라에 있어서 θ , ϕ , f 가 변경될 경우에 그에 따른 \bar{H} , \bar{V} , \bar{A} , \bar{C} 를 계산하는 원리와 본 시스템에서 이동과 회전에 이를 응용한 방법에 대해 설명하고자 한다.

2.1 카메라 설정

일단 C.O.P(center of projection)와 카메라의 \bar{A} 가 주어지면 나머지 \bar{H} 와 \bar{V} 는 자동적으로 구해진다. 그림 1에 극좌표계에서 카메라의 위치와 방위를 나타내었다.

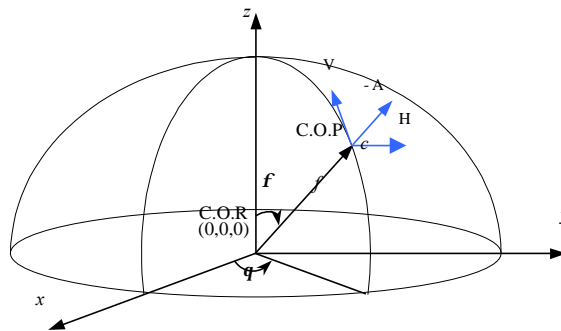


그림 1 극좌표계에서의 가상카메라

여기서 C.O.P는 다음과 같이 표현된다.

$$C = (C_x, C_y, C_z) = (f \cos \theta \sin \phi, f \sin \theta \sin \phi, f \cos \theta) \quad (1)$$

카메라 프레임 좌표인 \bar{H} 와 \bar{V} 는 방위각 θ 와 극각 ϕ 의 방향에서의 C.O.P의 접선을 의미한다. 그러므로 \bar{H} 와 \bar{V} 는 다음과 같이 ϕ 에 대해 C.O.P를 미분하면 구해진다.

$$\overline{H} = \frac{\partial C}{\partial \theta} = (-f \sin \theta \sin \phi, f \cos \theta \sin \phi, 0) \quad (2)$$

$$\overline{V} = \frac{\partial C}{\partial \phi} = (f \cos \theta \cos \phi, f \sin \theta \cos \phi, -f \sin \phi) \quad (3)$$

카메라 원점은 프로그램 내에서 조정되며, θ 와 ϕ 는 요(yaw)와 피치(pitch)로 표시된다. 원점에서 카메라까지의 거리도 역시 조정된다. 그렇지만 카메라 거리의 변경은 2D 이미지 평면에 투영되는 이미지의 적정크기가 보장되도록 자동적인 크기 조절을 수반해야 한다.

2.2 카메라 회전 및 이동

2.2.1 회전

그림 2에서 보는 바와 같이 카메라의 회전 시에 설정된 ϕ 와 θ 값인 $\text{cam}[n].\text{RotP}$, $\text{cam}[n].\text{RotT}$ 로부터 C.O.R(Center of Rotation)을 focusing 했을 때의 \overline{H} 와 \overline{V} 프레임을 계산한 후에 이동을 위해 현재 설정된 ϕ 와 θ 값인 $\text{cam}[n].\text{movP}$, $\text{cam}[n].\text{movT}$ 에 의해 이를 회전시킨다.

□입력값

Tilt angle	$\phi = \text{cam}[n].\text{MovP}$
Pan angle	$\theta = \text{cam}[n].\text{MovT}$
Focus distance	$f = \text{cam}[n].f$
C.O.P	$C = \text{cam}[n].C[0,1,2] : \text{Position}$

□출력값

C.O.R	$R = \text{cam}[n].R[0,1,2] : \text{Position}$
Camera direction (approach)	$A = \text{cam}[n].A[0,1,2] : \text{Unit vector}$
Horizontal camera frame	$H = \text{cam}[n].H[0,1,2] : \text{Unit vector}$
Vertical camera frame	$V = \text{cam}[n].V[0,1,2] : \text{Unit vector}$

$$C = (f \cos \theta \sin \phi, f \sin \theta \cos \phi, f \cos \phi) - R \quad (4)$$

$$\overline{A}_0 = |C| \quad (5)$$

$$\overline{H}_0 = (-f \sin \theta \sin \phi, f \cos \theta \sin \phi, 0) \quad (6)$$

$$\overline{V}_0 = (f \cos \theta \sin \phi, f \sin \theta \cos \phi, -f \cos \phi) \quad (7)$$

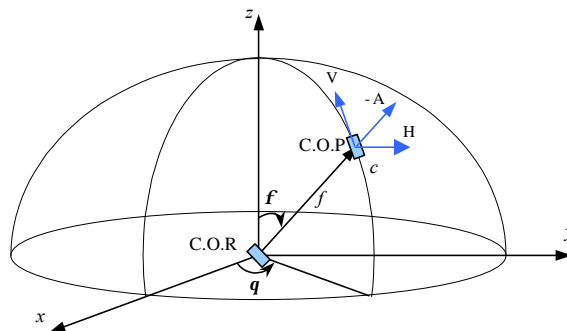


그림 2 카메라의 회전

2.2.2 이동

그림 3에서 보는 바와 같이 초기 \overline{H} - \overline{V} 프레임을 정하고 카메라 이동값인 $\text{cam}[n].\text{mov}_h$, $\text{cam}[n].\text{mov}_v$ 에 의해 \overline{H} , \overline{V} 방향으로 C.O.P를 이동시킨다.

$$C = (f \cos \theta \sin \phi, f \sin \theta \cos \phi, f \cos \phi) - R \quad (8)$$

$$\overline{A}_0 = |C| \quad (9)$$

$$\overline{H}_0 = (-f \sin \theta \sin \phi, f \cos \theta \sin \phi, 0) \quad (10)$$

$$\overline{V}_0 = (f \cos \theta \sin \phi, f \sin \theta \cos \phi, -f \cos \phi) \quad (11)$$

$$\overline{H} = \overline{H}_0 + \text{cam}[n].\text{mov}_h \quad (12)$$

$$\overline{V} = \overline{V}_0 + \text{cam}[n].\text{mov}_v \quad (13)$$

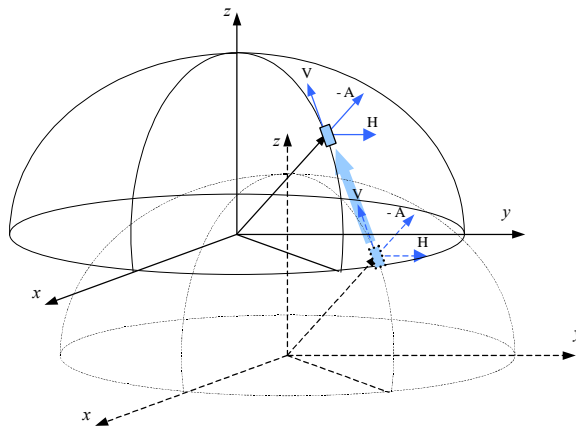


그림 3 카메라의 이동(수직방향)

3. 작업대상 정리

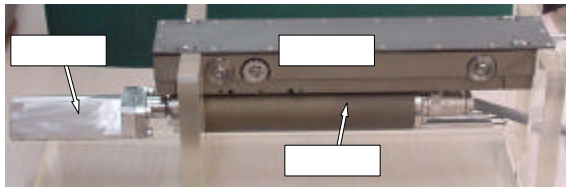
3.1 460 SG 슬러지세정장비와 작업내용

증기발생기 내부의 세관표면에 발생하는 슬러지(성분 Iron, Fe_3O_4 , Cu, NiO, ZnO, etc.)를 제거하기 위해 일반적으로 소형 로봇을 투입하여 노즐을 통해 세관사이로 고압수를 분사하는 방법을 사용한다. 지금까지는 외국의 장비를 고가에 들여와 작업을 진행하였는데 전력연구원에서는 그림 4(a)와 같은 세정로봇을 제작하여 국산화를 이루었다. 로봇 앞쪽의 노즐은 대각선 방향으로 배열되어 있으며 노즐모듈이 회전하면서 고압수를 세관사이로 분사하여 노즐 표면에 점착된 슬러지를 제거하고 반대쪽의 핸드홀에 설치된 별도의 흡입관을 통해 슬러지가 포함된 물을 증기발생기 외부로 배출하게 된다. 세관 세정 작업중에 관건이 되는 부분인 초기위치를 정하는 작업은 세정로봇의 제1노즐과 같은 위치에 있는 소형카메라에 의한 화상으로 로봇을 미세조정하여 이루어지고 초기위치가 정해지게 되면 가시화통을 이용하여 작업하게 된다.

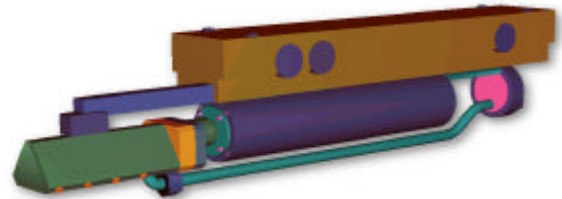
3.2 그래픽모델 생성

본 논문에서는 3차원 모델작성 및 렌더링 할 수 있는 패키지인 Autodesk사의 3ds max R3을 이용하여 가시화시스템에 사용할 로봇과 작업대상이 되는 증기발생기 일부분의 모델을 여러개의 오브젝트로 분리하여 개발하였다. 따라서 각 오브젝트에 독립적인 운동을 부여할 수 있게 되어 매우 사실적이고 세밀한 시뮬레이션이 가능하다. 그림 4(a)에 제작된 세정로봇을 나타내었고 그림

4(b)에 세정로봇의 그래픽 모델을 나타내었다. 이렇게 생성된 모델을 DirectX 패키지에서 제공되는 파일타입변환기로 DirectX에서 직접 읽어들이 수 있는 x파일로 변환하였다.



(a) 제작된 세정로봇



(b) 세정로봇 그래픽 모델

그림 4 세관세정로봇

4. 가시화 시스템 구축

4.1 윈도우 구성

그림 5는 DirectX를 이용하여 Visual C++로 작성된 가시화틀인 KeproSim이며 처음 실행시 작업영역을 모두 표시하도록 설정하였다. 그림 6은 로봇의 노즐부에 가상 카메라를 부착하여 정밀하게 세관사이에 위치시킨 모습이다. 정확한 위치를 파악하기 위해 노즐부를 실제와는 다르게 디자인 하였다. 각 메뉴에 대한 기능은 다음과 같다.

4.1.1 File

단계적실행 / 계속실행을 선택하고, 비트맵 버튼과 조인트 정보를 on/off 시키고, DirectX에서 이용 가능한 드라이브를 출력하며, 시스템을 종료하는 등의 메뉴들로 구성되어 있다. 이러한 기능들은 DirectX개발도구에서 제공되는 모든 프로그램들에서 동일한 사용자 인터페이스로 구성되어 있으며 본 시스템에서도 이를 이용하였다.

4.1.2 Render

Shade 모드인 flat / gouraud / phong을 선택하고, 오브젝트의 시각적 사실감을 증대시키는 light기능, fill 모드를 선택하는 point / wireframe / solid 기능, 오브젝트의 미세한 질감을 조절하는 dithering / anti-alias 기능, texture 필터를 선택하는 point filtering / bi-linear filtering 기능을 제공한다. 이들 기능도 File 메뉴와 마찬가지로 DirectX에서 제공되는 인터페이스를 사용하였다.

4.1.3 Camera

가상의 카메라를 선택하는 기능을 제공한다. 여기에는 로봇에 대해 가상 카메라의 상대적인 위치에 따라 top / right / front / left / bottom 의 5개의 카메라가 있으며, 이들 중에 하나를 선택할 수 있다. 만약 가상의 카메라가 더 필요하다면 프로그램의 코드를 일부 수정하여 추가할 수 있다.

4.1.4 Focus

로봇을 회전시킬 때 적용되는 원점(O.C.R)을 설정한다. 여기에는 로봇에 대해 원점의 상대적인 위치에 따라 auto / origin / loco / tilt 의 3개의 원점이 있으며 이들 중에 하나를 선택할 수 있다.

4.1.5 View

로봇의 특정부분을 선택적으로 화면에 보이게 하고, 나머지는 보이지 않게 하고 싶을 때 이용하는 기능이다. 여기서는 윈도우에 표시되는 물체들을 S/G / robot / nozzle / U-tube 의 4개 부분으로 나누었으며, 처음 실행 시에는 모든 오브젝트가 선택 되도록 하였으며, 작업자가 필요에 따라 일부 혹은 전부를 선택할 수 있다. 단 이 기능을 활용하기 위해서는 모델 작성시에 각 부분

을 별도의 모델로 작성해야 한다.

4.1.6 가상카메라 제어버튼

DirectX에서는 디스플레이 카드의 메모리에 존재하면서 이미지를 저장하는 표면(surface)을 지원한다. 따라서 여기서는 이를 이용하여 윈도우의 우측에 전체 윈도우의 약 1/5 크기로 back buffer surface를 생성하고, 여기에 BitBlt 및 StretchBlt 함수를 이용하여 툴바를 대신할 수 있는 제어 버튼을 그림 5와 같이 제작하였다. 본 버튼들은 팝업 메뉴들에 의해 설정된 정보를 토대로 물체를 이동시키고, 회전시키며, 축소 및 확대를 수행하는 기능을 제공한다. 이들 제어메뉴는 윈도우가 축소/확대되더라도 크기가 변하지 않으며 항상 메인 윈도우의 우측상단에 고정되어 있다.

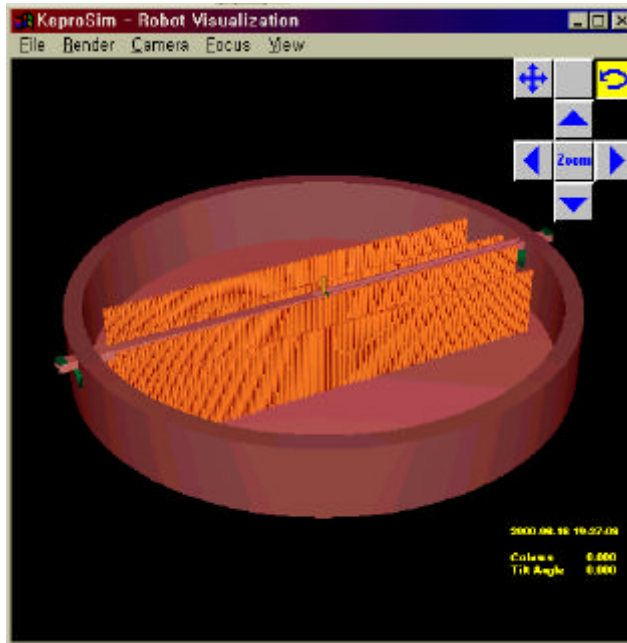


그림 5 메인 윈도우의 구성

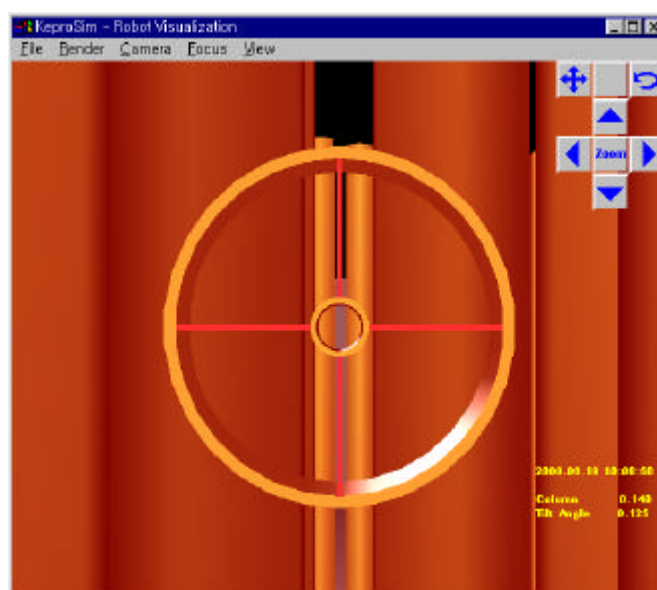


그림 6 작업과정(작업위치 정밀조작)

4.2 로봇언어

개발된 세정장비에 명령을 주는 언어는 크게 3가지 방법으로 입력이 가능하다. 먼저 모터제어기(LM628)가 장착된 KALANS 서보에서 키보드로 직접 명령을 주는 방법이 있고, 파일에 저장된 명령을 순차적으로 수행하는 방법이 있으며, 마지막으로 주제어 컴퓨터에서 컴퓨터 통신에 의해 원격으로 명령을 주는 방법이 있다. 입력된 명령은 번역기로 입력되어 주어진 로봇언어에 해당하는 명령을 코드화하고 명령의 수행에 필요한 데이터를 입력한 후 이동경로생성 프로그램으로 보내져서 로봇 이동이 수행된다. 로봇언어에 오류가 발생한 경우 이를 CRT에 표시하거나 경고음으로 사용자에게 알리는 기능도 포함되어 있다. 로봇언어는 적용되는 작업에 따라 code를 변경함으로써 간단하게 추가/제거가 가능하다. 표 1에 본 시스템에서 사용된 로봇언어 명령을 나타내었다.

표 1 로봇언어 명령

로봇언어	기 능
home	move to zero position
speed	percent of full speed
loco	move(mm)
tilt	rotate(deg.)
column	(mm)
reset	zero for all motors
delay	(sec.)
stop	
file	setup motion file
pump	pump on/off
defh	define home

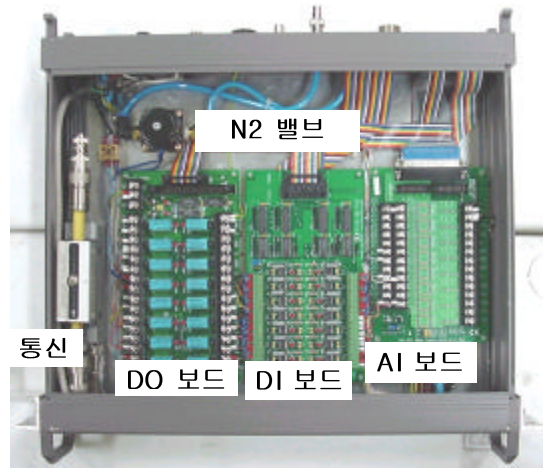
5. 제어시스템 제작

5.1 제어기 하드웨어

KALANS(Kepco Advanced LANcing System; 한진 전력연구원에서 제작한 증기발생기 세정 로봇의 명칭) 제어/감시시스템은 크게 증기발생기 세관 세정로봇을 원격으로 제어 및 감시하는 KALANS 마스터와 로봇을 구동하는 KALANS 서보로 구성된다. KALANS 마스터는 격납용기 외부의 제어실에 위치하고 서보는 격납용기 내부에 설치되며 이들은 컴퓨터 통신(UDP/IP)에 의해 정보를 교환한다. KALANS 마스터는 그림 7과 같이 제어용컴퓨터, 주제어기, Video Switch, VTR로 구성되어 있으며 주제어기는 Digital I/O보드, Analog input보드, 로봇에 장착된 카메라의 시야를 확보하기 위해 질소를 분사하게 하는 질소밸브, 그리고 KALANS 서보와 통신하기 위한 UDP/IP Transceiver로 구성되어 있다. KALANS 마스터는 사용자 인터페이스를 위한 로봇원격제어와 감시환경을 제공한다. KALANS 서보는 그림 8과 같이 2축 모터제어기(LM628)와 모터 구동장치인 Power Amplifier, 제어명령을 전달받아 제어지령을 내리는 컴퓨터, 그리고 모터 과전류 보호기로 구성되어 마스터로부터의 로봇제어명령을 수행하고 로봇제어상태를 전달한다. KALANS 제어/감시시스템의 전체 구성은 아래의 그림 9와 같다.



(a) KALANS Master 외관

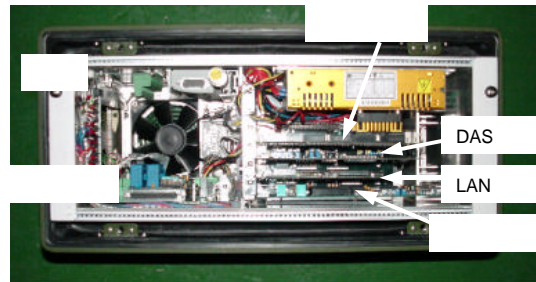


(b) KALANS Master 주제어기 내부구조

그림 7 KALANS Master



(a) KALANS Servo 외관



(b) KALANS Servo 내부구조

그림 8 KALANS Servo

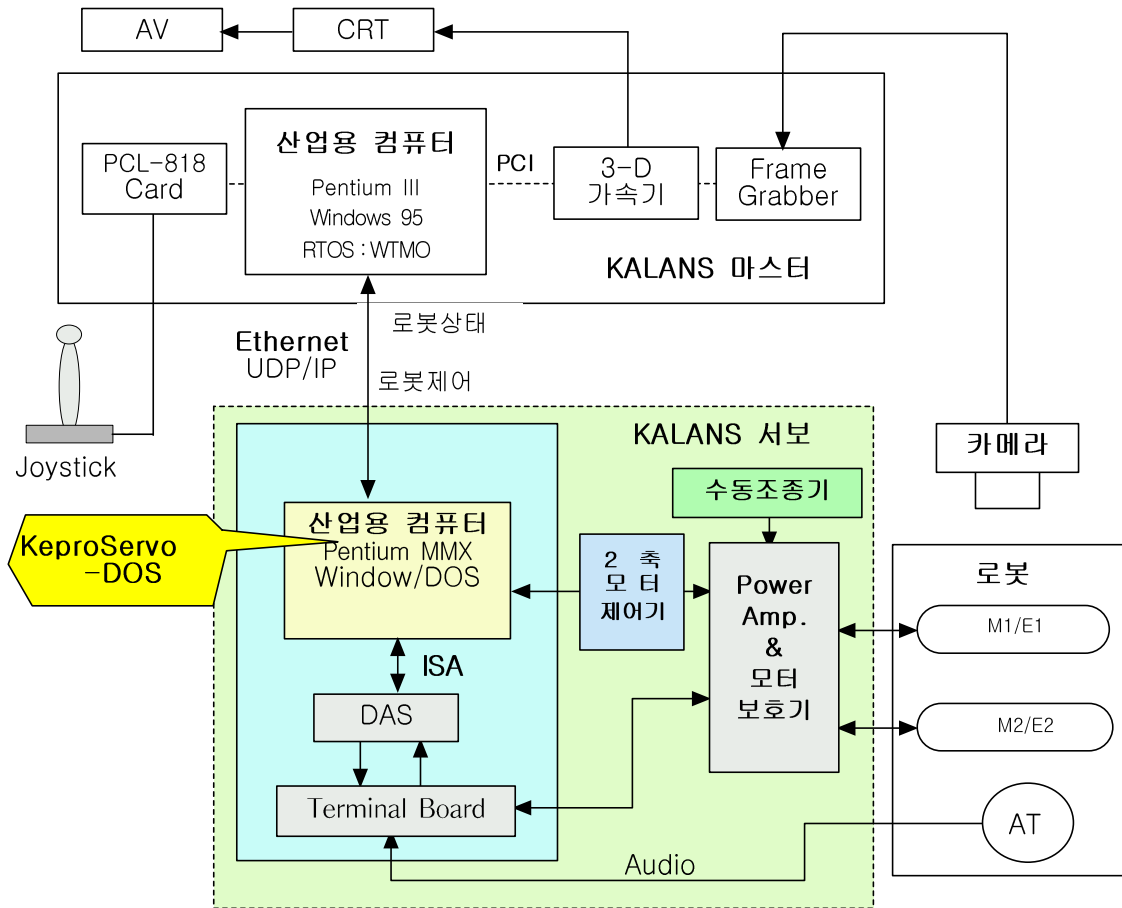
그림 9의 KALANS Servo에 탑재되어 있는 KeproServo-DOS는 서보의 컴퓨터 보드에 장착되어 있는 플래쉬롬에 설치되어 있는 시스템 소프트웨어이며 전원이 켜지면 자동으로 실행된다. KeproServo-DOS는 주로 마스터와의 통신에 의해 로봇의 제어와 감시를 수행하지만, 기본적으로 KALANS Master에 설치되어 있는 KeproTeleOp와 동일한 사용자 인터페이스를 가지고 있기 때문에 서보만으로도 키보드와 로봇언어 파일에 의해 로봇의 자동제어가 가능하도록 구현되어 있다. 서보시스템에서 수행하는 작업은 다음과 같다.

□ 실시간 작업

- 모터제어기 구동 및 상태감시
- 모터전원, 자동/수동 상태, 과전류 릴레이 동작상태 감시
- 실시간 프로그램 수행관리

□ Background 작업

- 주제어기로부터의 컴퓨터 통신 또는 축제어 컴퓨터에서 사용자에게 의해 입력된 로봇언어의 번역 및 해당 로봇 이동경로 생성
- 로봇 운전현황과 상태를 CRT에 표시
- 주컴퓨터와의 통신
- 사용자가 입력하는 인터럽트신호의 처리



註) M: Motor, E: Encoder, B: Braker, L: Limit Switch,
 AV: Audio/Video System, AT: Acoustic Transmitter
 그림 9. KALANS 로봇 시스템의 구성

5.2 제어기 소프트웨어

KALANS 제어/감시 시스템의 소프트웨어는 총 4가지로 구성되어 있으며 각각 사용자 인터페이스, 실시간 원격제어/감시를 수행하는 KeproTeleOp(그림 10참조)와 로봇의 거동을 Online으로 Visualization하거나 Offline으로 시뮬레이션하는 KeproSim, 로봇에 부착된 카메라에 의해 수신되는 작업 이미지를 캡션과 함께 모니터에 표시하는 KeproCam 및 KeproTeleOp 프로그램으로부터 제어지령을 통신을 통해 전달받아 이를 이용하여 로봇을 제어하는 KeproServo로 구성된다. KeproTeleOp, KeproSim, KeproCam은 KALANS Master부의 제어용 컴퓨터에서 실행되며 제어 신호를 주제어기에 전송하게 된다. KeproServo는 KALANS Servo의 컴퓨터보드내에서 실행되며 제어명령을 수행하는 역할과 로봇의 상태를 KALANS Master에 통신을 통해 전달하는 역할을 담당하게 된다. KeproTeleOp, KeproSim과 KeproRobot는 전력연구원이 등록한 로봇원격제어/감시 소프트웨어의 명칭이다. 이들은 모두 MS Visual C++에서 윈도우 환경으로 개발되었으며 코드의 수정으로 다양한 로봇제어 분야에 활용할 수 있다. 이들 소프트웨어의 주요특징으로는 여러기능을 모듈화하여 시스템의 안정성을 향상시켰고 시스템 관리를 용이하게 하였으며, 로봇시스템 개발 운용 및 사용자 훈련을 위한 통합환경을 마련하였고, Windows와 산업용 PC 환경을 채용하여 시스템 정비와 사용자 편이성이 좋은 경제적인 Solution임을 들 수 있다.그림 11은 이들 프로그램들의 기능, 처리절차와 데이터의 흐름을 보이고 있다.

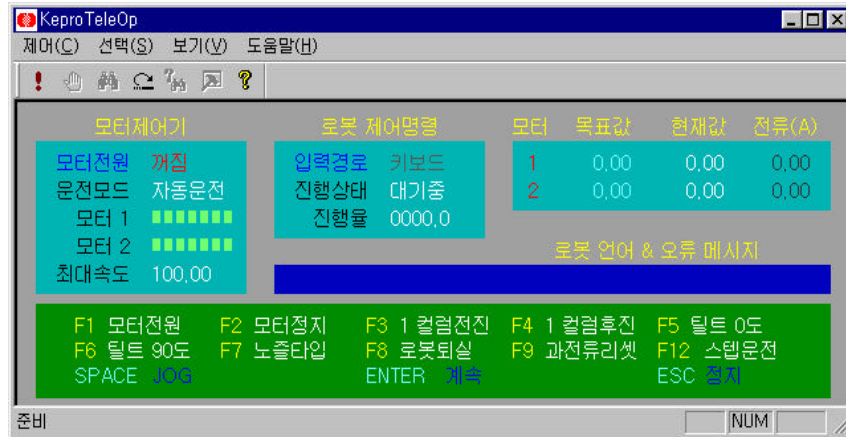


그림 10 KeproTeleOp 화면

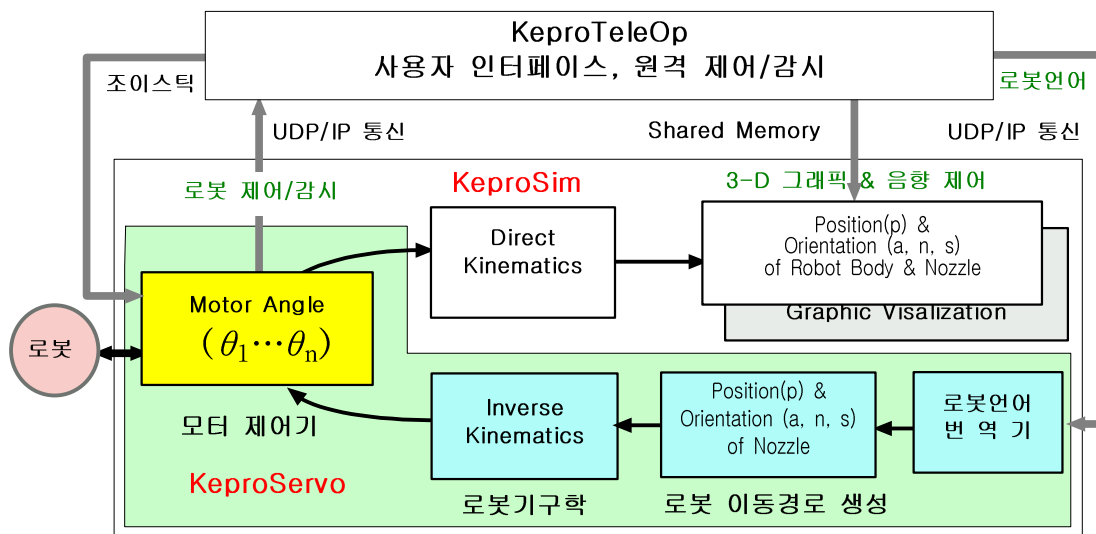


그림 11 KALANS 제어/감시 소프트웨어의 구성

본래 본 프로그램 다축로봇제어를 염두에 두고 제작 되었기 때문에 프로그램 내에서 기구학적 해석을 통해 제어명령을 수행하게 되며 그래픽처리 또한 기구학적 해를 이용해 수행한다. 동작원리를 살펴보면, 우선 자동운전의 경우 사용자가 UDP/IP통신을 통해 서보에 명령을 내리면 로봇언어 번역기에서 로봇 End-effector의 최종위치와 자세(Orientation)을 생성하고 이것의 역기구학을 취하면 각 모터의 회전각도가 얻어진다. 다관절 로봇에 적용할 경우 역기구학의 해가 유일하지 않으므로 로봇이 취해야 하는 Configuration (Upper/Lower Arm, Right/Left Arm, Wrist Flip 등)을 미리 정하여야 한다. 모터제어기는 이 회전각도를 목표로 주어진 속도와 가속도로 회전하게 된다. 모터가 회전하면 현재의 회전상태가 마스터에 알려지고, KeproSim의 Visualization 프로그램은 로봇 정기구학(Direct Kinematics) 수식을 풀어 현재의 로봇위치와 자세를 계산하고 이것을 3차원으로 그래픽처리를 하게 된다. 서보와 마스터 사이의 정보교환은 UDP/IP 컴퓨터통신으로 이루어지고 마스터에서 수행되는 프로그램 모듈들은 Shared Memory 방식으로 이루어진다. 로봇의 위치와 자세 정보는 Shared Memory를 통해 KeproTeleOp에서 KeproSim으로 전달되고, 사용자는 KeproSim을 통해 로봇의 움직임을 3차원 그래픽으로 볼 수 있다. 그림 12는 제어기 조작부로서 실험용 Mock Up을 이용하여 작업하는 모습이다.



그림 12 제어기 조작부

6. 결 론

본 논문에서는 460형 증기발생기 슬러지세정작업의 3차원 가시화를 위해 DirectX를 이용한 로봇 visualization 시스템과 로봇제어기 제작에 관하여 기술하였다. 본 시스템에서는 다수의 가상카메라를 설치하여 로봇 작업내용을 실시간으로 다양한 각도와 위치에서 관찰할 수 있으므로 작업 능력과 완성도에 있어 획기적인 개선을 이루었다. 이는 지난해 수행된 고리1발전소의 계획예방정비기간 동안 증기발생기 세관세정작업에 투입된 결과 입증되었으며 460형과 F형 증기발생기에 모두 사용가능 하도록 수정한 후 올해 5월로 예정된 고리2발전소의 계획예방정비 기간에 세관세정작업에 투입될 예정이다. 아울러 본 visualization 시스템은 개방된 구조로서 작업모델과 이에 따른 파라미터를 쉽게 추가, 변경할 수 있으므로 여러 가지 작업에 활용될 수 있는 특징을 가지고 있다. 또한 제어기에 있어서 모든 제어기를 방사능 구역인 작업장소에 설치하지 않고 최소장비만 설치하기 때문에 작업중 고장수리의 어려움이 줄어들었으며 작업이후 제염 해체과정도 상당부분 용이하게 진행할 수 있다. 향후 원자로 내부의 육안검사용 수중로봇의 Visualization 제어시스템 구축에 본 시스템을 활용할 계획이다.

참고문헌

- [1] 박신열, 허태영, "3D Studio와 DirectX를 이용한 Visualization 시스템 개발," 한국 정보처리학회 추계학술대회논문집, pp.CG-43~CG48, 1999. October
- [2] 조병학, "로봇의 기구학적 3차원 그래픽 시뮬레이션," 한국시뮬레이션학회 추계학술대회 논문집, pp.202~209, 1999. October
- [3] 조병학, "Simulation Studies on Robot System Applied to Nozzle Dam Installation," 전력연구원, TM. 96EW03.P1998.482, 1998
- [4] 한전전력연구원, "460형 증기발생기 슬러지 세정장비 개발," KEPRI TM.99NJ02.M2000.155, 2000.4.
- [5] 한전전력연구원, "정비지침서 (KALANS 감시 제어 시스템)," KEPRI TM.99NJ02.P2000.436, 2000.10.
- [6] Microsoft, "Microsoft Direct3D Retained Mode," Microsoft, 1998 annal