



경고

모든 설치 작업은 반드시 자격있는
설치기사에 의해 수행되어야 하며
관련 법규 및 규정을 준수하여야 합니다.





Hi6 제어기 기능설명서

팔레타이즈





본 제품 설명서에서 제공되는 정보는 현대로보틱스의 자산입니다.
현대로보틱스의 서면에 의한 동의 없이 전부 또는 일부를 무단 전재 및 재배포할 수 없으며,
제 3 자에게 제공되거나 다른 목적에 사용할 수 없습니다.

본 설명서는 사전 예고 없이 변경될 수 있습니다.

Printed in Korea - 2023 년 4 월. 2 판
Copyright © 2023 by Hyundai Robotics Co., Ltd



목 차

1. 개요	1-1
1.1. 현대로보틱스 로봇 팔레타이징 S/W	1-2
1.2. 특징	1-3
2. HRpal v2.1	2-1
2.1. HRpal 흐름도	2-2
2.2. 팔레트, 패키지, 간지 정보	2-3
2.2.1. 팔레트 정보	2-3
2.2.2. 패키지, 간지 정보	2-4
2.3. 패턴 생성	2-5
2.3.1. 패턴 생성 메뉴	2-5
2.3.2. 패턴 생성 결과	2-7
2.4. 패턴 수정	2-9
2.4.1. 적재물의 위치 수정	2-10
2.4.2. 순서 변경	2-12
2.4.3. 추가	2-13
2.4.4. 방향전환	2-13
2.4.5. 삭제	2-15
2.4.6. 팔레트 좌표계 원점 설정	2-16
2.4.7. 핸드 개방 방향 설정	2-17
2.4.8. 저장	2-17
2.5. 패턴 관리	2-18
2.5.1. JOB 생성	2-19
2.5.2. 옵션 수정	2-19
2.5.3. 패턴 삭제, 패턴 저장, 종료	2-20
2.6. 사용자 패턴 정의	2-21
3. 팔레타이징 표준 프로그램	3-1
3.1. 변수 구성	3-2
3.2. 전체 순서도	3-5
3.3. 프로그램 구성	3-6
3.3.1. 메인 프로그램	3-7
3.3.2. Vacuum On, Off	3-9
3.3.3. 공정 파라미터 설정	3-9
3.3.4. 각종 변수, 신호 초기화	3-11
3.3.5. 홈포지션, 작업물 위치 포즈 프로그램	3-12
3.3.6. 톨 제어 프로그램	3-12
3.3.7. 팔레트별 적재패턴, 간지 시프트 변수 프로그램	3-13
3.3.8. 팔레트 좌표계 생성 프로그램	3-14
3.3.9. 팔레트, 간지 파라미터 설정	3-16

3.3.10. 팔레타이징, 간지 모션 프로그램.....	3-18
3.4. 표준 프로그램의 기본적인 사용.....	3-20
3.4.1. 변수 설정.....	3-20
3.4.2. 위치 기록	3-20



그림 목차

그림 1.1 HRpal v2.1 메인화면.....	1-2
그림 2.1 HRpal 사용 흐름도.....	2-2
그림 2.2 팔레트 정보 메뉴.....	2-3
그림 2.3 패키지 정보 메뉴.....	2-4
그림 2.4 간지 정보 메뉴.....	2-4
그림 2.5 패턴 생성기 메뉴.....	2-5
그림 2.6 단처짐 보상 기능 적용 예시.....	2-6
그림 2.7 패턴 생성 결과 메뉴 구성.....	2-7
그림 2.8 패턴 수정 메뉴.....	2-9
그림 2.9 패턴 수정 중, 상호 간섭이 발생하는 경우.....	2-10
그림 2.10 적재물의 무게 중심이 팔레트 영역 밖으로 이동할 경우.....	2-11
그림 2.11 적재 순서 변경 기능.....	2-12
그림 2.12 적재물 추가 기능.....	2-13
그림 2.13 적재물 방향 전환 기능.....	2-13
그림 2.14 적재물 삭제.....	2-15
그림 2.15 팔레트 좌표계 원점 설정 기능.....	2-16
그림 2.16 팔레타이징용 그리퍼들.....	2-17
그림 2.17 핸드 개방 방향 지정 기능.....	2-17
그림 2.18 패턴 관리 기능.....	2-18
그림 2.19 옵션 수정 메뉴.....	2-19
그림 2.20 사용자 패턴 정의 메뉴.....	2-21
그림 3.1 표준 프로그램 전체 순서도.....	3-5
그림 3.2 팔레타이징 작업 시 사용되는 좌표 정보.....	3-9
그림 3.3 사용자 좌표계 기록 예시.....	3-14
그림 3.4 오른손 좌표계의 구성 형태.....	3-15
그림 3.5 팔레타이징 동작 위치값.....	3-18

표 목차

표 3-1 표준 프로그램 내 processInfo 변수 구성 및 용도	3-2
표 3-2 표준 프로그램 내 simulator 변수 구성 및 용도	3-3
표 3-3 표준 프로그램 내 pos 변수 구성 및 용도	3-3
표 3-4 표준 프로그램 내 시프트 변수 번호 및 용도	3-4
표 3-5 표준 프로그램 JOB Number 및 역할	3-6





HD

HYUNDAI
ROBOTICS

1

개요



1. 개요

1.1. 현대로보틱스 로봇 팔레타이징 S/W

로봇 팔레타이징은 일정한 위치에서 작업물을 집어 팔레트 또는 랙(Rack)등에 내려놓는 동작을 반복 수행하는 작업입니다.

효율적인 로봇 팔레타이징 시스템을 구성하기 위해서는 로봇의 이동경로를 고려한 컨베이어, 팔레트 등을 설치해야 하며, 이에 따라 라인의 생산성이 달라집니다.

HRpal™ v2.1은 현대로보틱스 로봇 팔레타이징 적재패턴 및 Job 프로그램을 자동 생성할 수 있는 PC(HRpal) 기반 프로그램으로 빠른 설치 시운전 검토를 통해 효율적인 로봇 팔레타이징 시스템을 구성을 가능하게 합니다.



그림 1.1 HRpal v2.1 메인화면

1.2. 특징

- 팔레타이징 적재 최적패턴 자동 생성 기능 (사용자 패턴정의 가능)
- 팔레타이징 표준 Job 프로그램 생성 기능
- 팔레타이징 적재패턴 관리기능







HD

HYUNDAI
ROBOTICS

2

HRpal v2.1



3. HRpal v2.1

팔레타이즈

2.1. HRpal 흐름도

HRpal 은 PC 기반 팔레타이징 응용 소프트웨어 입니다. 추가적으로 HRSpace3 시뮬레이터용 프로젝트 파일을 만들어주는 기능이 추가되어 있습니다.

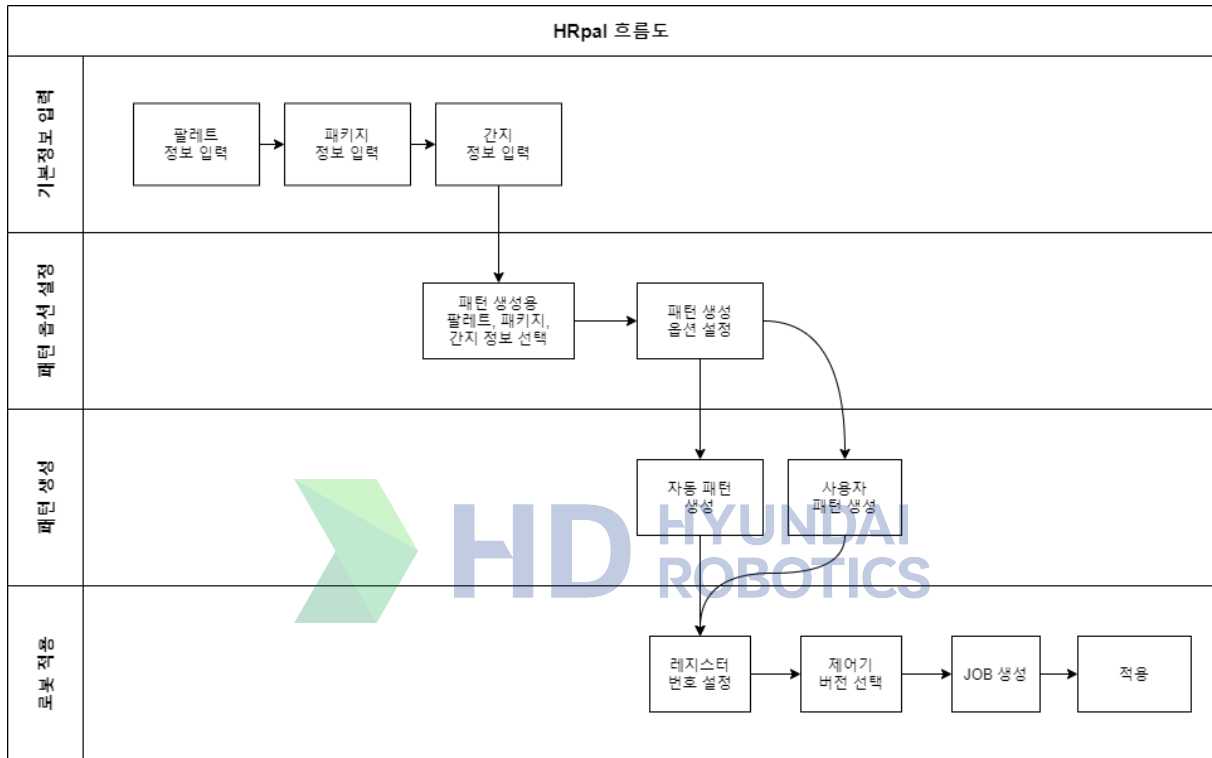


그림 2.1 HRpal 사용 흐름도

2.2. 팔레트, 패키지, 간지 정보

2.2.1. 팔레트 정보

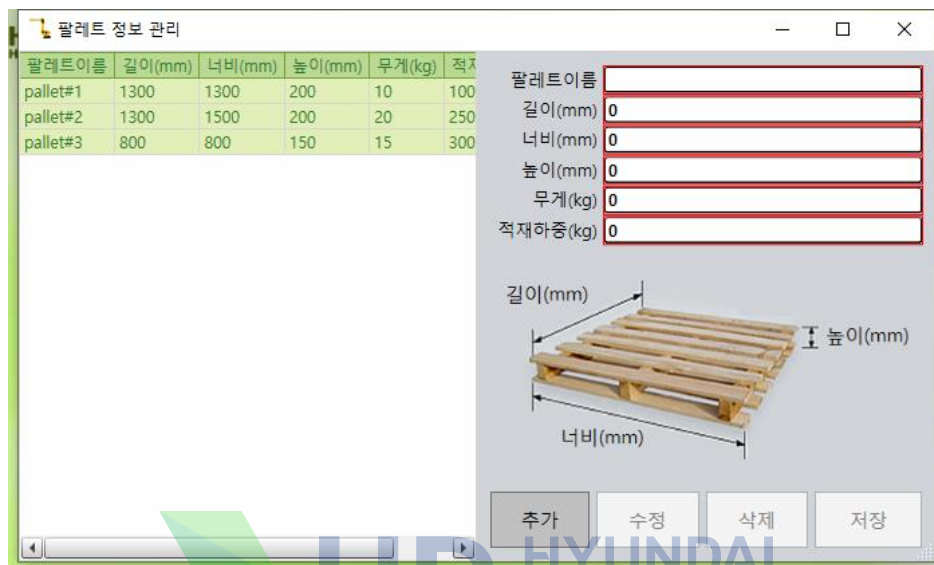
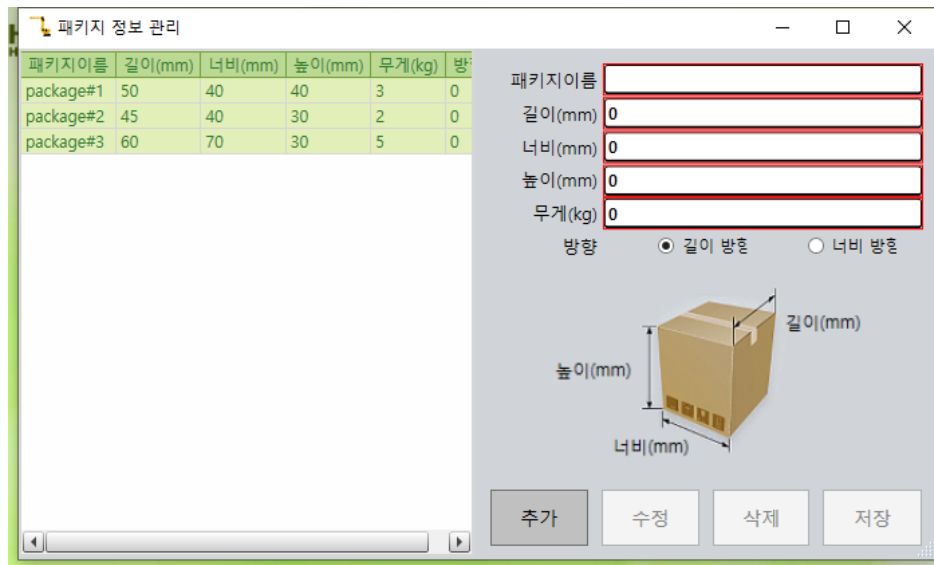


그림 2.2 팔레트 정보 메뉴

팔레트 정보는 이름, 길이, 너비, 높이, 무게, 적재하중으로 구성되어 있습니다(그림 2-2). 무게는 팔레트 자체의 무게를 의미하고, 적재하중은 팔레트가 견딜 수 있는 최대 하중을 의미합니다. 그리고 화면 하단부의 버튼 메뉴는 추가, 수정, 삭제, 저장이 있으며 각각의 기능은 다음과 같습니다.

- (1) 추가
추가 버튼은 새로운 데이터를 추가하고자 할 때 사용합니다. 모든 데이터가 입력되어 있을 경우에 정상적으로 활성화 됩니다.
- (2) 수정
수정 버튼은 왼쪽의 리스트가 선택되어 있을 경우에 활성화됩니다. 리스트 항목 중 한가지를 선택하면 내용이 오른쪽에 나타나게 됩니다. 그리고 해당 내용을 수정한 후, 수정 버튼을 누르면 자동으로 해당 항목이 수정되어 파일로 저장됩니다.
- (3) 삭제
삭제 버튼도 수정 버튼과 마찬가지로 왼쪽의 리스트 항목이 선택되면 활성화됩니다. 수정버튼과는 달리 삭제의 경우 파일로 자동으로 저장되지 않습니다.
- (4) 저장
저장 버튼은 삭제 수행 후, 해당 내용을 저장하려 할 때 사용합니다.

2.2.2. 패키지, 간지 정보



패키지이름	길이(mm)	너비(mm)	높이(mm)	무게(kg)	방
package#1	50	40	40	3	0
package#2	45	40	30	2	0
package#3	60	70	30	5	0

패키지이름:

길이(mm):

너비(mm):

높이(mm):

무게(kg):

방향: ☒ 길이 방향 ☐ 너비 방향

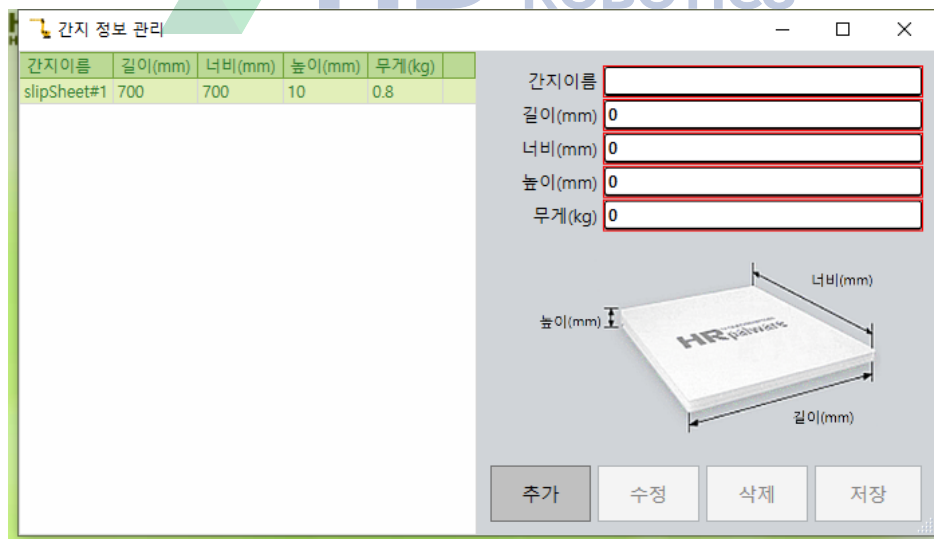
높이(mm)

길이(mm)

너비(mm)

추가 수정 삭제 저장

그림 2.3 패키지 정보 메뉴



간지이름	길이(mm)	너비(mm)	높이(mm)	무게(kg)
slipSheet#1	700	700	10	0.8

간지이름:

길이(mm):

너비(mm):

높이(mm):

무게(kg):

높이(mm)

너비(mm)

길이(mm)

추가 수정 삭제 저장

그림 2.4 간지 정보 메뉴

패키지(그림 2-3)와 간지 정보(그림 2-4)는 이름, 길이, 너비, 높이, 무게 항목을 입력하여 사용합니다. 기타 나머지 사용 방법은 팔레트 정보와 동일합니다.

2.3. 패턴 생성

2.3.1. 패턴 생성 메뉴

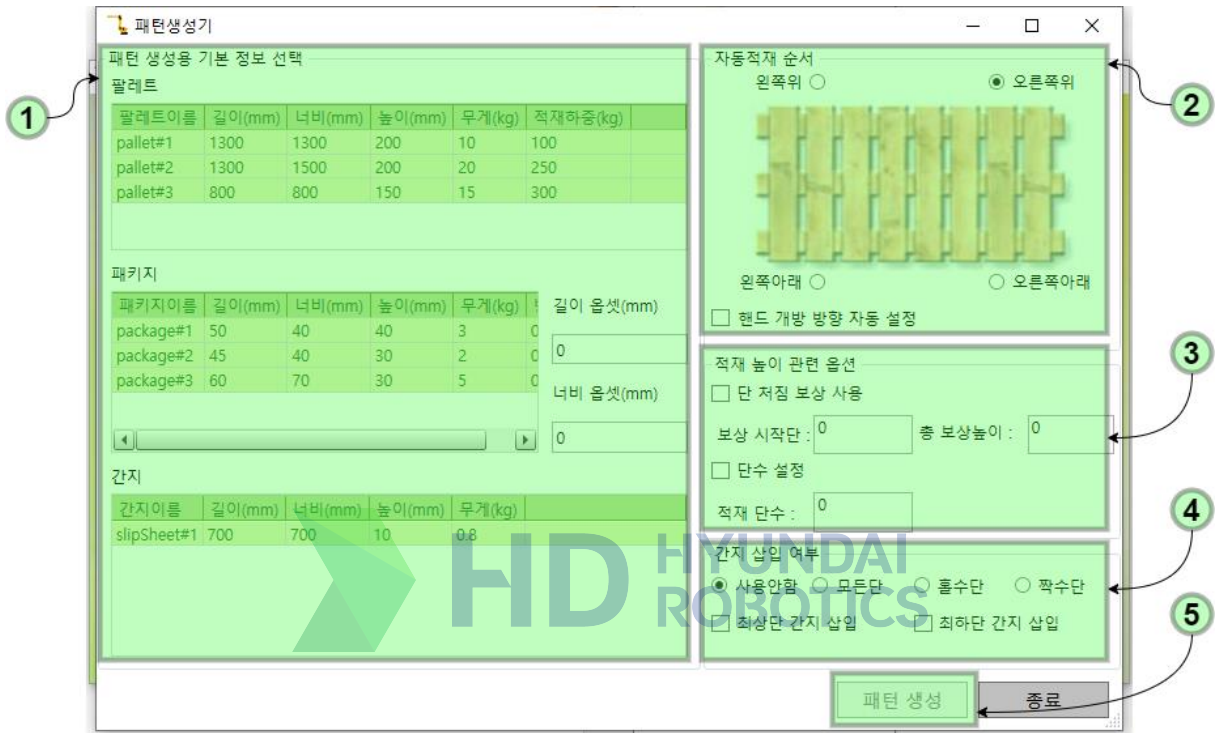


그림 2.5 패턴 생성기 메뉴

패턴 생성 메뉴는 팔레트, 패키지, 간지 정보를 사용하여 최적화된 팔레타이징 패턴을 만들어주는 역할을 합니다(그림 2-5). 각 메뉴의 설명은 다음과 같습니다.

(1) 패턴 생성용 기본 정보 선택

앞서, 설명한 팔레트, 패키지, 간지 정보에서 추가한 내용들이 나타나 있으며, 패턴을 만들고자 하는 대상 물체를 클릭하여 선택합니다.

패키지의 길이, 너비 옵션은 각 물체 별로 일정한 간격을 두고 싶을 때 사용합니다.

(2) 자동 적재 순서

자동 적재 순서는 팔레트를 기준으로 하여 어느 위치부터 팔레타이징을 수행할 지 결정한다. 기본적으로 오른쪽위로 선택되어 있습니다.

(3) 적재 높이 관련 옵션

HRpal 은 팔레트의 적재하중을 기반으로 하여 단수를 결정하는데, 단수 설정 옵션은 이를 무시하고, 사용자가 입력한 수치의 단수를 사용하는 옵션 입니다.

단수 설정은 3 단, 혹은 4 단과 같이 정확한 단수가 정해져 있을 때 사용합니다.

단 처짐 보상은 Bag 팔레타이징과 같이 물건이 쌓일수록 아래로 처지는 현상이 나타나는 팔레타이징에 대응하기 위한 옵션입니다.

단처짐 기능은 그림 2-6 와 같이 보상 시작 단부터 이후 나머지 단까지, 총 보상 높이를 나눈 만큼 분배하여 적용 됩니다.

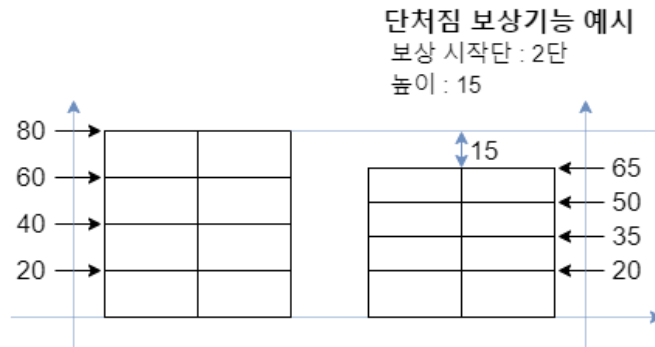


그림 2.6 단처짐 보상 기능 적용 예시

(4) 간지 삽입 여부

간지 옵션은 사용안함, 모든단, 홀수, 짝수단이 선택 가능합니다.

최하단 간지 삽입 옵션은 작업 시작 전에 간지를 바닥에 까는 행위를 수행할 지를 결정합니다.

최상단 간지 삽입 옵션은 마지막 모든 작업이 끝난 후, 간지로 팔레트를 덮는 행위를 수행할지를 결정합니다.

(5) 패턴 생성

패턴 생성에 필요한 모든 옵션을 선택하면 패턴 생성 버튼이 활성화 됩니다. 그리고 패턴 생성 버튼을 누르면 해당 데이터를 토대로 최적화된 적재 패턴을 만들고, 그림 2-7 과 같은 화면이 나타납니다.

2.3.2. 패턴 생성 결과

패턴 생성 결과 메뉴에서는 사용자가 선택한 데이터를 토대로 최적화된 패턴과 몇 가지 다른 형태의 패턴을 보여줍니다(그림 2-7). 사용자는 해당 패턴을 그대로 사용하고자 할 때에는 저장 버튼을 눌러 데이터를 저장한 후, 추후 패턴 관리 메뉴를 통해 해당 패턴을 JOB 프로그램 형식으로 추출할 수 있습니다.

패턴 생성 결과 메뉴의 각 부분별 기능은 다음과 같습니다.

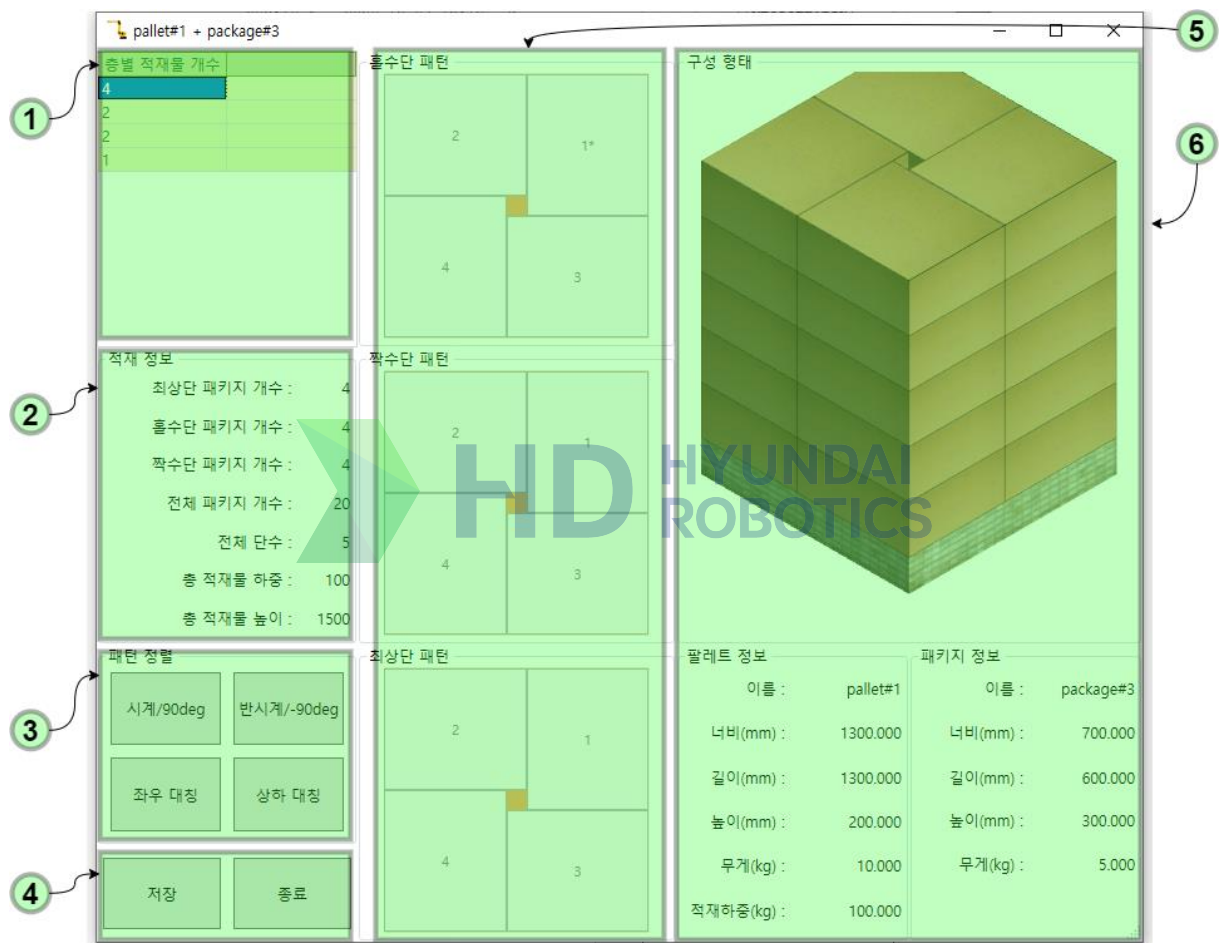


그림 2.7 패턴 생성 결과 메뉴 구성

(1) 패턴 리스트

사용자가 선택한 데이터를 토대로 만들어진 패턴 결과물의 리스트를 보여줍니다.

선택한 데이터에 따라 적재 정보 및 패턴 형태, 구성 형태 등의 내용도 함께 변환됩니다.

⑤의 최상단, 홀수단, 짝수단 중 특정 단을 선택한 후, 다른 리스트를 선택하면 해당 부분만 변경됩니다.

(주의) 홀, 짝수단 적재물 개수보다 최상단의 적재물 개수가 많은 경우, 나머지 홀, 짝수단의 적재물 개수도 많은 쪽 개수에 맞춰지도록 되어 있습니다.

(2) 적재정보

적재 정보는 선택한 데이터의 보다 구체적인 하중, 높이 등을 나타냅니다.

(3) 패턴 변경

해당 패턴을 90 도 방향으로 움직이거나 좌, 우 대칭 시키고자 할 때 이용합니다.

그림 3.6 의 ⑤에 있는 최상단, 홀수단, 짝수단 중 하나를 선택한 후 패턴을 변경하면, 해당 부분만 변경됩니다.

(4) 저장, 종료

패턴을 저장하고 종료하는데 사용합니다.

(5) 층 별 패턴 데이터

현재 선택된 데이터들의 홀, 짝수단, 그리고 최상단의 패턴 데이터를 보여줍니다.

특정 단의 패턴 데이터를 수정할 때에는 마우스를 사용하여 더블 클릭하시면 됩니다.

수정에 대한 자세한 내용은 [2.4. 패턴 수정]에 서술되어 있습니다.

(6) 구성 형태

현재 패턴 데이터의 예상 구성 형태를 3D 로 구성하여 나타냅니다.

마우스 좌, 우 버튼을 사용하여 보이는 각도를 조절할 수 있습니다.

아래쪽에는 현재 생성된 패턴을 만드는데 사용된 팔레트 및 패키지의 정보를 나타냅니다.



2.4. 패턴 수정

패턴 수정 메뉴(그림 2-8)는 아래와 같은 기능을 가지고 있으며, 홀수단, 짝수단, 최상단 중 특정 패턴을 더블 클릭하여 열 수 있습니다.

- (1) 적재물의 위치를 수정
- (2) 적재 순서의 변경
- (3) 적재물 추가 및 제거
- (4) 적재물의 방향 전환
- (5) 핸드 개방 방향 설정
- (6) 팔레트 좌표계 원점 설정

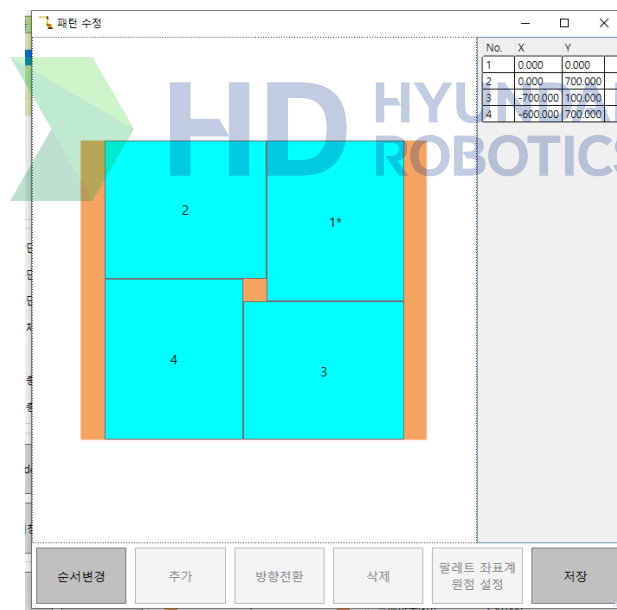


그림 2.8 패턴 수정 메뉴

2.4.1. 적재물의 위치 수정

적재물의 위치는 다음의 세 가지 방법으로 변경이 가능합니다.

- (1) 적재물을 마우스로 선택하여 드래그 & 드롭으로 위치를 변경
- (2) 적재물을 선택한 후, 키보드 화살표 키로 위치를 변경
- (3) Grid 창에서 X, Y 를 직접 입력하는 방법

만약 위치 변경 중에 작업물 간의 상호 간섭이 발생할 경우, 아래의 (그림 2-9)과 같이 상호 간섭이 발생한 물체들이 붉게 표시됩니다.

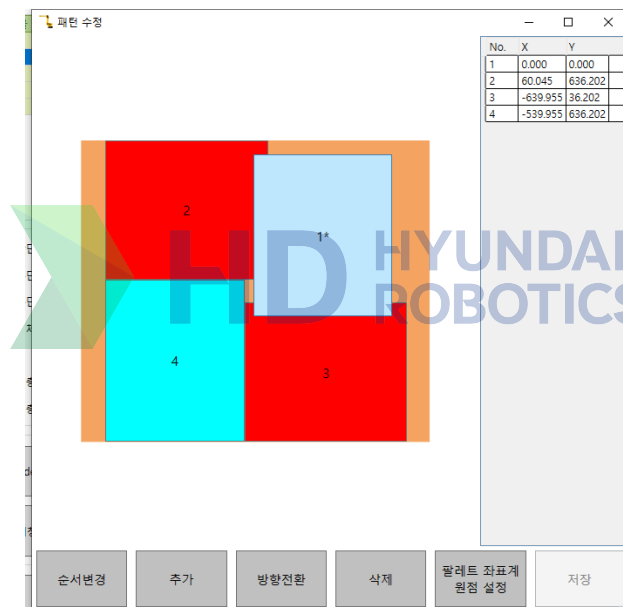


그림 2.9 패턴 수정 중, 상호 간섭이 발생하는 경우

또한, 팔레트 밖으로도 상자를 배치할 수 있는데, 이 경우에는 적재물의 중심점이 팔레트 영역 밖으로 나갈 경우, 붉게 표기합니다(그림 2-10).

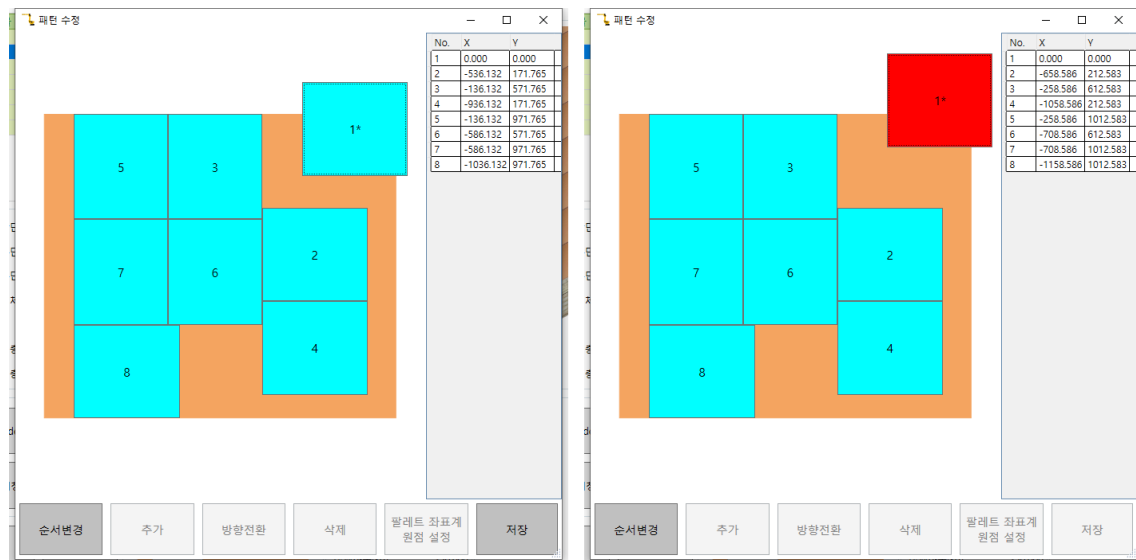


그림 2.10 적재물의 무게 중심이 팔레트 영역 밖으로 이동할 경우

2.4.2. 순서 변경

순서 변경은 적재물의 적재 순서를 변경하고자 할 때 사용합니다. 순서 변경 버튼을 클릭하면, 아래의 그림 2-11의 (1)과 같이 모든 적재물의 적재 순서가 초기화되며, 마우스로 적재하고자 하는 순서대로 클릭하면 그림 2-11의 (2)와 같이 상자의 색이 노란색으로 변하면서 번호가 지정됩니다.

만약 수정을 위해 적재물을 순서대로 클릭하던 도중 일부를 다시 수정하고자 할 때는, 해당 번호의 상자 클릭 시 그 상자과 이후에 입력한 정보는 초기화되어 다시 입력할 수 있습니다. 예를 들어 1번부터 5번까지의 순서가 입력된 상태에서 3번 선택 시 3, 4, 5번 순서는 지워지고, 3번째 순서부터 다시 입력할 수 있습니다.

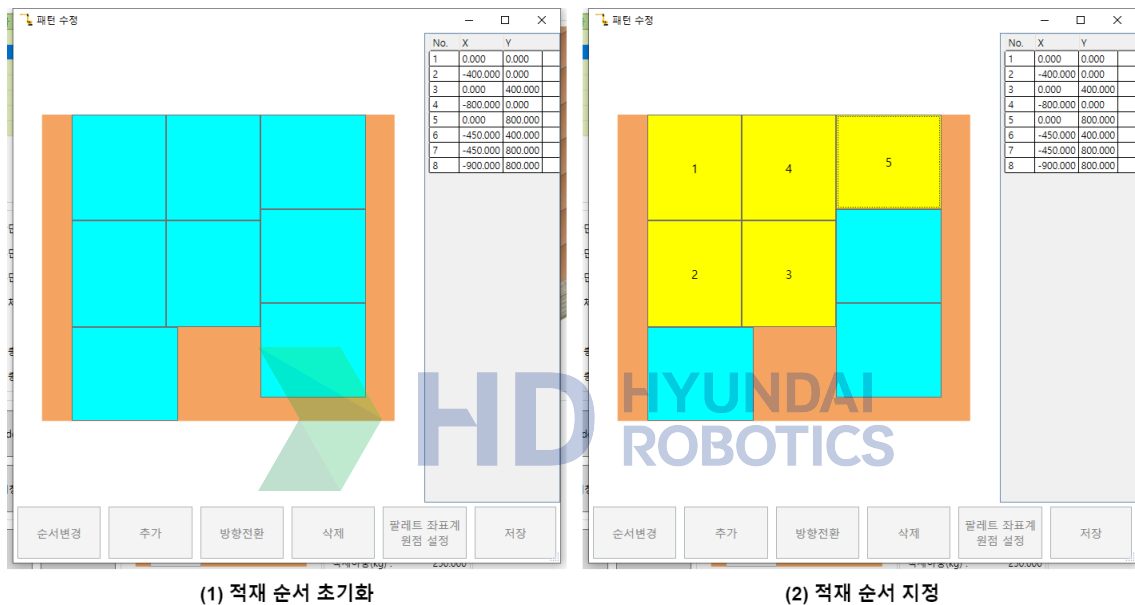


그림 2.11 적재 순서 변경 기능

2.4.3. 추가

추가를 원하는 형태의 상자를 선택한 후, 추가버튼을 누르면, 선택된 상자와 동일한 형태의 상자가 추가됩니다(그림 2-12).

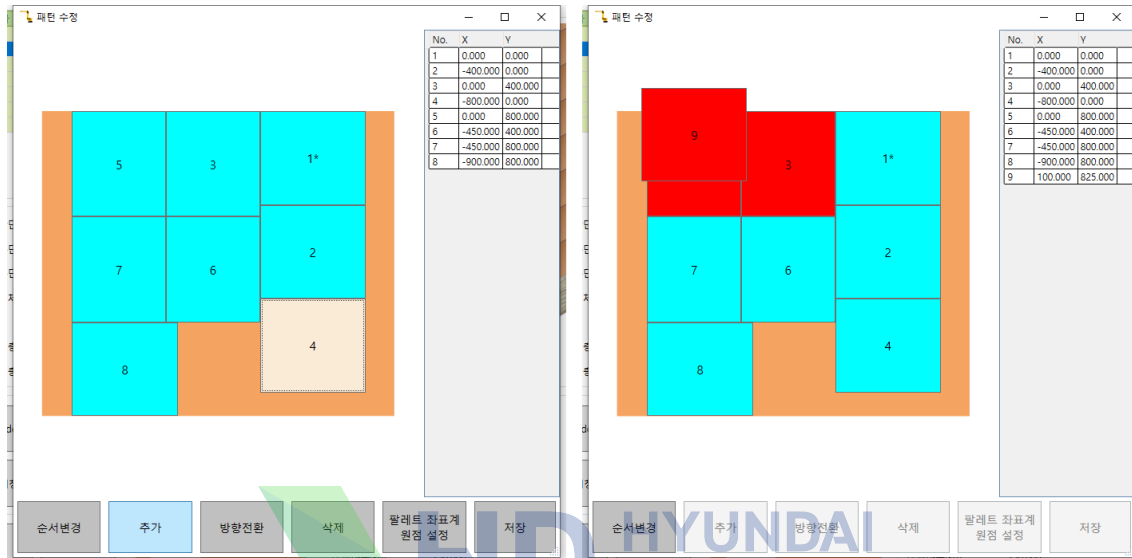


그림 2.12 적재물 추가 기능

2.4.4. 방향전환

방향 전환은 선택된 상자를 가로 또는 세로로 방향을 전환하고자 때 사용합니다. 아래의 그림 2-13 은 4 번 상자를 선택하여 방향전환을 한 후, 위치를 재배치한 결과 입니다.

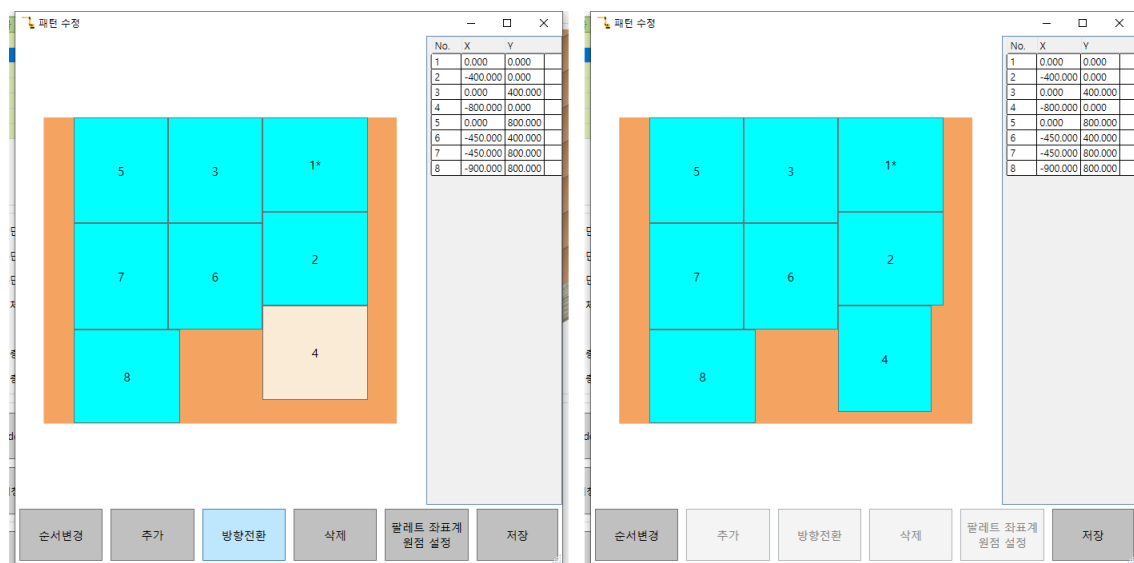


그림 2.13 적재물 방향 전환 기능



2.4.5. 삭제

삭제는 패턴 내에서 특정 적재물을 삭제하고자 할 때 사용됩니다. 삭제 하고자 하는 작업을 선택한 후, 삭제 버튼을 누르면 해당 적재물이 사라지게 됩니다. 이때, 삭제한 적재물보다 적재 순서가 더 뒤에 있는 적재물들은 적재 순서가 재배치 됩니다(그림 2-14).

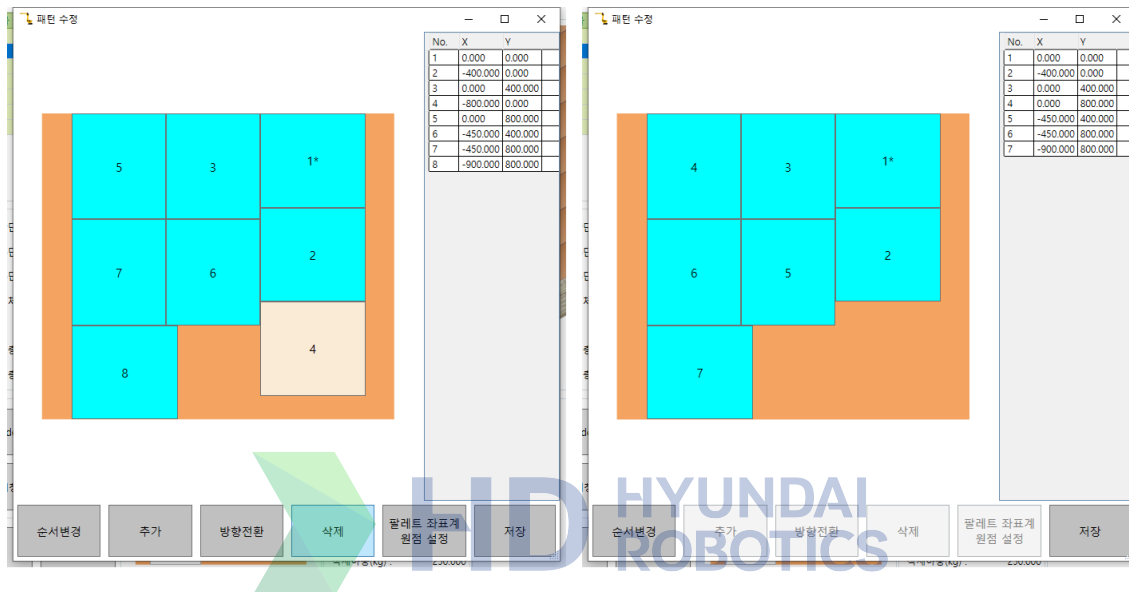


그림 2.14 적재물 삭제

2.4.6. 팔레트 좌표계 원점 설정

HRpal 을 사용하여 생성되는 표준 JOB 프로그램에서 적재 패턴은 팔레트 좌표계를 기준으로 하여 생성됩니다. 팔레트 좌표계는 사용자가 정의하는 팔레트 상에서의 좌표계를 의미하며, 아래의 그림 3.14 에 나타나있는 오른손 모양과 동일한 좌표계를 사용합니다.

팔레트의 위치나 방향, 형태는 매번 달라질 수 있기 때문에 사용자가 직접 팔레트 상에서 원점과 X 축 방향 그리고 Y 축 방향에 대해서 티칭 해주어야 합니다. 팔레트 좌표계 원점은 팔레트 좌표계를 티칭 할 때 기준이 되는 점을 의미합니다. 기본적으로 홀수단 1 번 작업물을 기준으로 하지만, 사용자의 편의에 의해 팔레트 좌표계 원점 설정 기능을 통하여 다른 작업물을 원점으로 설정할 수 있습니다.

팔레트 좌표계 원점을 변경하고자 하는 작업물을 선택한 후, 팔레트 좌표계 원점 설정 버튼을 클릭하면 해당 위치의 작업물에 * 표시가 나타나고, 원점이 변경됩니다(그림 2-15).

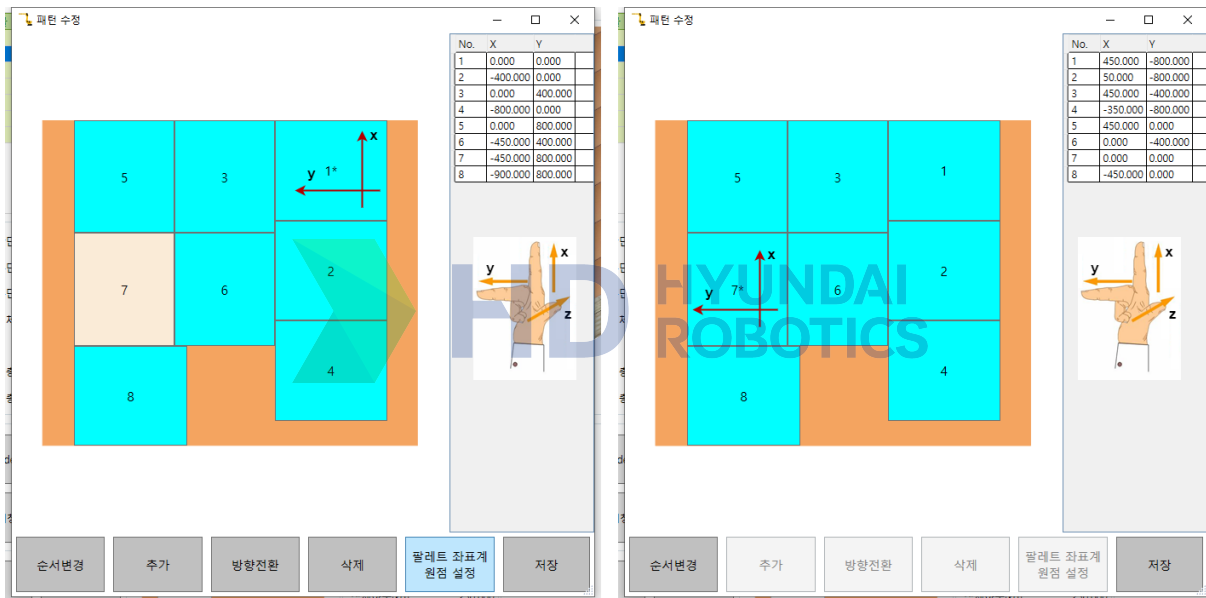


그림 2.15 팔레트 좌표계 원점 설정 기능

2.4.7. 핸드 개방 방향 설정

팔레타이징용 그리퍼는 아래와 같이 방향의 구분이 필요 없이 위에서 내려놓는 방식이나, 특정 방향으로 핸드가 개방 방향을 고려 해야 하는 것이 있습니다(그림 2-16). 핸드 개방 방향 지정 기능은 이러한 개방 방향을 고려해야 하는 핸드를 위해 방향을 지정할 수 있는 기능입니다.



그림 2.16 팔레타이징용 그리퍼들

각각의 적재물에 대해 마우스 오른쪽 버튼을 누르면 방향 지정 키가 활성화 되며, 상, 하, 좌, 우 중 아무 곳이나 마우스를 이동하면 자동으로 지정됩니다(그림 2-17). 또한, 방향성을 지정하지 않기를 원하면, 오른쪽 버튼을 누른 후, 바로 왼쪽버튼으로 중앙을 선택하면 됩니다.

여기서 화살표는 화살표의 방향으로 핸드가 개방된다는 것을 의미합니다. 그리고 지정된 데이터를 사용하여 추후 JOB 프로그램 생성시에 핸드의 회전 방향을 계산하여 저장합니다.

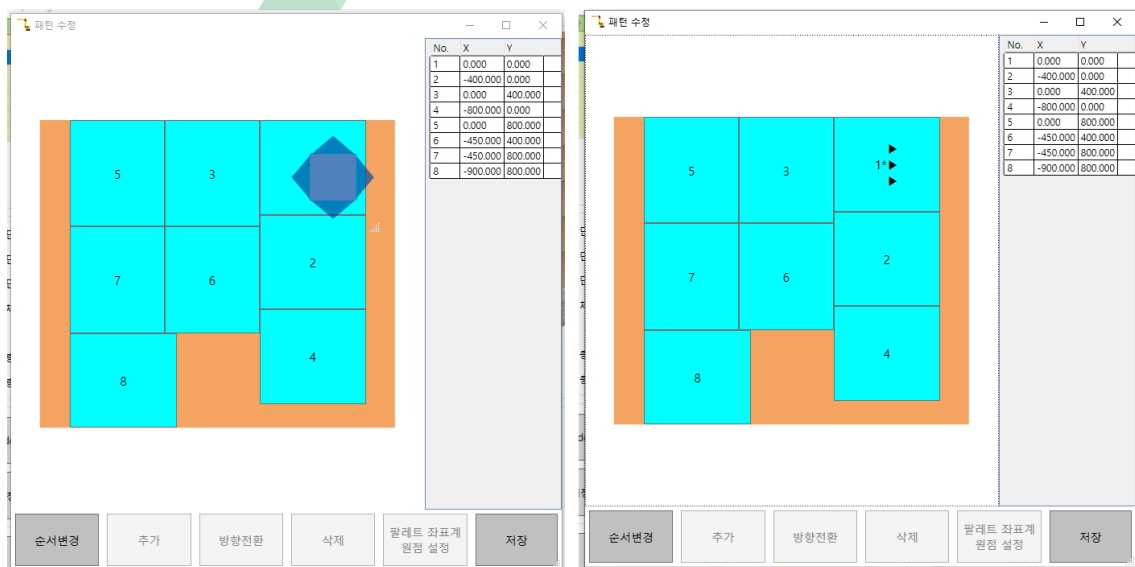


그림 2.17 핸드 개방 방향 지정 기능

2.4.8. 저장

저장 버튼은 위치의 변경, 삭제 등 패턴에 대한 수정이 이루어진 후, 실제 패턴에 반영하기 위해 사용됩니다. 저장 버튼을 누르면 자동으로 해당 창이 종료되며, 선택했던 단의 내용이 변경된 것을 확인할 수 있습니다.

2.5. 패턴 관리

패턴 관리는 사용자가 기존에 만들었던 패턴들을 관리하기 위한 메뉴 입니다(그림 2-18). 그리고 로봇에서 사용하기 위한 JOB 프로그램을 만들거나, POSE 파일을 생성하기 위해서 사용할 수 있습니다. 패턴 레지스터는 JOB 프로그램에서 사용될 해당 패턴에 대한 번호이며, 직접 입력하여 지정할 수 있습니다. 그리고 핸드 방향 설정이 필요한 패턴에 대해서는 해당 옵션을 지정하여 사용할 수 있습니다.

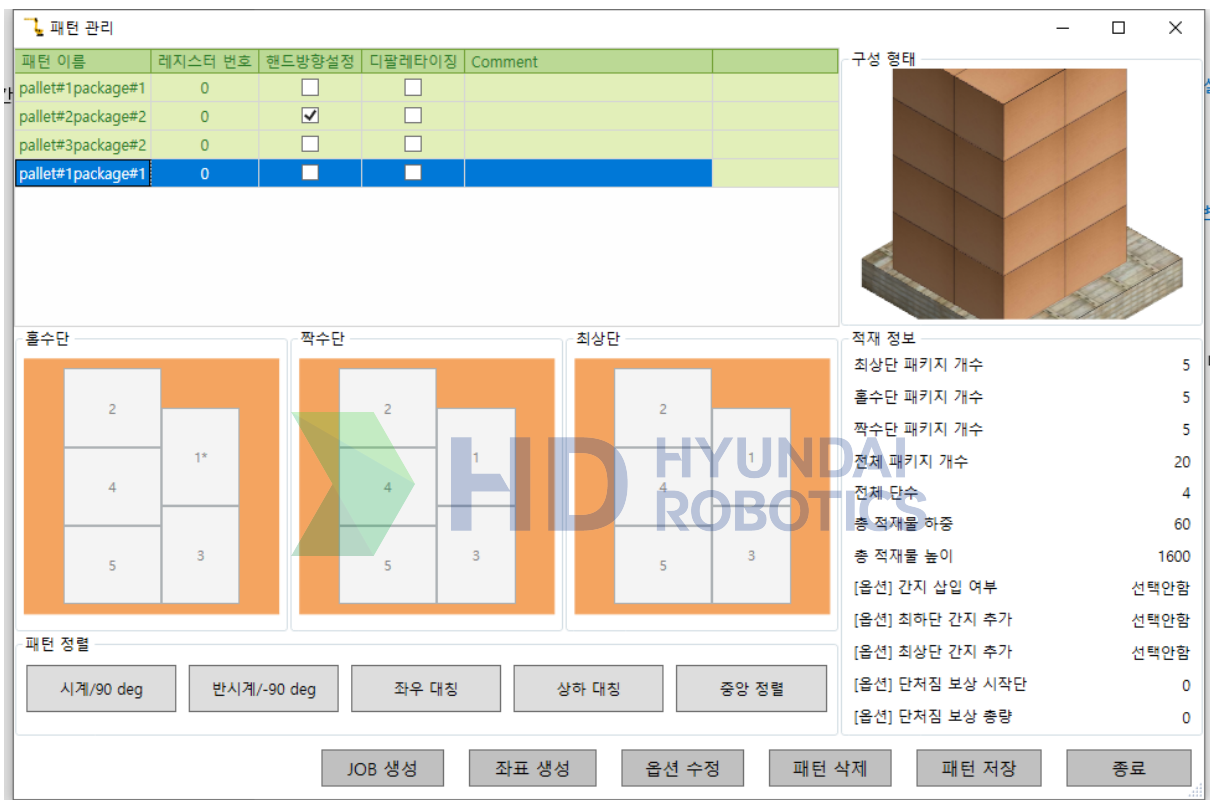


그림 2.18 패턴 관리 기능

2.5.1. JOB 생성

JOB 생성 기능은 레지스터 번호가 지정된 패턴에 대해 바로 팔레타이징 기능을 수행해볼 수 있도록 표준 JOB 프로그램1과 HRSpace3 시뮬레이션 환경을 생성하여 출력하는 기능입니다. 해당 기능의 특징은 아래와 같습니다.

- 패턴 레지스터 번호를 할당한 패턴들을 실제 로봇에서 사용할 수 있도록 표준 프로그램을 생성하여 출력합니다.
- HRSpace3에서의 시뮬레이션 편의를 위해 해당 패턴에 대한 적재물 모델을 자동으로 생성해줍니다.
- 패턴 레지스터 번호는 1~16 까지 할당 가능하며, 중복 할당은 불가능 합니다.
- 핸드 방향 설정 옵션이 활성화 되어 있는 경우, 각 적재물의 핸드 개방 방향이 미지정 상태이면 JOB 프로그램을 생성할 수 없습니다.
- JOB 생성 버튼을 누르면 시뮬레이션을 수행하고자 하는 제어기의 버전을 선택하는 메뉴가 나타납니다.
- 제어기 버전 선택 이후, JOB 프로그램을 저장하고자 하는 폴더를 선택하는 메뉴가 나타납니다. 해당 폴더 선택 후, 확인 버튼을 누르면 해당 폴더에 JOB 프로그램이 저장됩니다.
- 만약 어떠한 패턴 레지스터도 설정되어 있지 않다면, 수행할 수 없습니다.
- 더 이상 JOB 프로그램을 생성하기를 원치 않는 패턴에 대해서는 패턴레지스터를 0 번으로 변경해야 합니다.

2.5.2. 옵션 수정

옵션 수정은 단 처짐과 관련된 보상 옵션과 간지 삽입 여부를 수정하기 위한 기능입니다(그림 2-19).

The dialog box titled '옵션 수정' (Option Modification) contains the following elements:

- 적재 높이 관련 옵션** (Load Height Related Option):
 - ☐ 단 처짐 보상 사용 (Use single sag compensation)
 - 보상 시작단 : (Compensation start stage)
 - 총 보상높이 : (Total compensation height)
- 간지 삽입 여부** (Interference Insertion):
 - ☒ 사용안함 (Do not use) | ☐ 모든단 (All stages) | ☐ 홀수단 (Odd stages) | ☐ 짝수단 (Even stages)
 - ☐ 최하단 간지 삽입 (Insert interference at the bottom) | ☐ 최상단 간지 삽입 (Insert interference at the top)
- Buttons at the bottom: **저장** (Save) and **종료** (End).

그림 2.19 옵션 수정 메뉴

¹ 표준 프로그램에 대한 자세한 설명은 “3. 팔레타이징 표준 프로그램”을 참고하세요.

2.5.3. 패턴 삭제, 패턴 저장, 종료

- 패턴 삭제 버튼은 현재 지정된 패턴을 삭제하는데 사용됩니다.
(주의) 현재 화면에서 패턴을 삭제하더라도, 패턴 저장 버튼을 누르지 않으면 파일에 반영되지 않습니다.
- 패턴 저장 버튼은 패턴의 일부 내용을 수정하거나 변경한 내용을 파일로 저장하고자 할 때 사용됩니다.



2.6. 사용자 패턴 정의

사용자 정의 기능은 사용자가 임의대로 적재물을 배치하여 패턴을 만들 수 있는 기능입니다(그림 2-20). 자동 패턴 생성 기능과 달리 서로 다른 적재물을 배치하여 패턴을 만들 수 있습니다. 기능의 사용 순서는 다음과 같습니다.

- (1) 바닥에 놓일 팔레트를 선택합니다.
- (2) 적재하고자 하는 적재물을 선택합니다.
- (3) 추가, 방향전환, 삭제 등의 기능을 사용하여 원하는 패턴을 구성합니다.
- (4) 시계, 반시계 방향등 패턴 편집 기능을 사용하여 패턴을 정 중앙에 위치시킵니다.
- (5) 짝수단, 최상단의 패턴도 위의 방법을 사용하여 정의합니다.
- (6) (*) 홀수단 복사 기능을 사용하여 홀수단과 동일한 패턴을 사용할 수 있습니다.
- (7) 적재단수를 지정합니다.
- (8) 기타 단처짐 보상이나 간지 삽입여부를 지정합니다.
- (9) 저장 버튼²을 누르고 저장할 패턴의 이름을 지정하면, 패턴 등록이 완료됩니다.

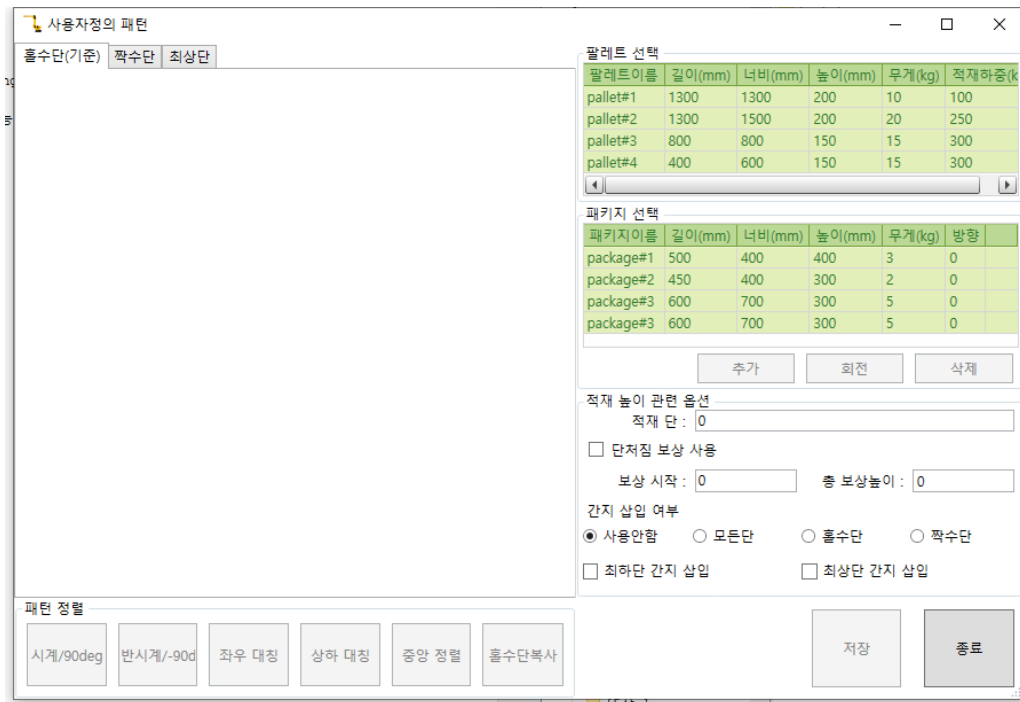


그림 2.20 사용자 패턴 정의 메뉴

² 저장 기능은 홀수단, 짝수단, 최상단의 데이터가 온전히 채워져 있고, 적재단수가 할당되어 있을 때 활성화 됩니다.





HD

HYUNDAI
ROBOTICS

3

팔레타이징
표준 프로그램



3. 팔레타이징 표준 프로그램

팔레타이즈

팔레타이징 표준 프로그램은 팔레타이징에 필요한 툴을 사전에 구성해놓은 프로그램 집합입니다. 사용자가 HRpal 을 사용하여 팔레타이징을 수행할 때, 해당 프로그램을 사용하면 작업 관련 위치의 기록 및 변수의 변경만으로 쉽고 빠르게 팔레타이징을 수행할 수 있습니다.

3.1. 변수 구성

표 3-1 표준 프로그램 내 processInfo 변수 구성 및 용도

processInfo 전역변수		
Object Array	16 개 Object 로 구성된 배열이며, 공정 관련 정보를 통합 관리하는 변수 정의되지 않은 공정은 0 으로 초기화되며, 예를 들어 3 번, 6 번 공정 번호로 Job 이 생성되 었다면 아래와 같이 구성됨 [0, 0, Object {regNo:3,} , 0, 0, Object {regNo:6,} , 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	
Type	Key	내용
Integer	regNo	공정 번호 저장 변수
Integer	totalPackages	해당 공정의 총 작업물 개수(간지 제외)
Integer	oddPackages	홀수단 작업물 개수
Integer	evenPackages	짝수단 작업물 개수
Integer	topPackages	최상단 작업물 개수
Integer	totalSlipsheet	해당 공정의 총 간지 개수
Boolean	firstLayerSlipSheet	최하단 간지 삽입여부
Boolean	topLayerSlipSheet	최상단 간지 삽입여부
Float	packageLength	작업물 크기 (L)
Float	packageWidth	작업물 크기 (W)
Float	packageHeight	작업물 크기 (H)
Float	pickupHeight	들어올릴 높이 (초기값은 packageHeight * 1.5)

표 3-2 표준 프로그램 내 simulator 변수 구성 및 용도

simulator 전역변수		
Object	Palletizing 시뮬레이션 관련 정보 저장 객체	
Type	Key	내용
Integer	currRegNo	현재 공정 번호
Integer	prevRegNo	직전 공정 번호
Integer	currActionNo	공정의 작업 번호 (0 ~ (총작업물수+총간지수)-1)
Integer	currPackageType	현재 적재물의 Type 번호 (1~5, 7 이면 간지로 판단)
Integer	prevPackageType	이전 적재물의 Type 번호 (1~5, 7 이면 간지로 판단)
Integer	cntPackage	작업물 Counter
Integer	cntSlipsheet	간지 Counter
Integer	cntLayer	단수 Counter
Pos Object	originUCS	사용자 좌표계 원점 좌표 (첫번째 작업물 목표 위치)
integer	error	시뮬레이션 도중 Error 발생 시 정보 저장 변수
Pos Object	homePos	홈포지션 위치
Pos Object	nearPickupPos	픽업 위치 바로 위
Pos Object	pickupPos	픽업 위치
Pos Object	nearTargetPos	적재 패턴 위치 바로 위
Pos Object	targetPos	적재 패턴 위치

표 3-3 표준 프로그램 내 pos 변수 구성 및 용도

pos 전역변수		
Object Array	16 개 Object 로 구성된 배열이며, 공정 관련 Position 정보 관리 변수 정의되지 않은 공정의 정보는 0 으로 초기화 되며, 예를 들어 3 번, 6 번 공정 번호로 Job 이 생성되었다면 아래와 같이 구성됨 [0, 0, Object {homePos:Pos Object(),}, 0, 0, Object {homePos:Pos Object(),}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	
Type	Key	내용
Pos Object	homePos	공정별 기본 자세(홈포지션)
Pos Object	pickupPos1	공정별 1 번 적재물을 들어올릴 위치 (공정 당 5 종류 적재물 사용 가능)
Pos Object	pickupPos2	공정별 2 번 적재물을 들어올릴 위치
Pos Object	pickupPos3	공정별 3 번 적재물을 들어올릴 위치
Pos Object	pickupPos4	공정별 4 번 적재물을 들어올릴 위치
Pos Object	pickupPos5	공정별 5 번 적재물을 들어올릴 위치

Pos Object	SlipSheetPos	공정별 간지를 들어올릴 위치
------------	--------------	-----------------

표 3-4 표준 프로그램 내 시프트 변수 번호 및 용도

Shift 전역변수		
Type	Key	내용
Shift Array	packageShift1	1 번 공정 적재물 시프트 변수 예시) packageShift1[1], packageShift1[55]
Shift Array	packageShift2	2 번 공정 적재물 시프트 변수
Shift Array	packageShift3	3 번 공정 적재물 시프트 변수
...		
Shift Array	packageShift16	16 번 공정 적재물 시프트 변수
Shift Array	slipSheetShift1	1 번 공정 적재물 시프트 변수 예시) slipSheetShift1[1], slipSheetShift1[55]
Shift Array	slipSheetShift2	2 번 공정 간지 시프트 변수
Shift Array	slipSheetShift3	3 번 공정 간지 시프트 변수
...		
Shift Array	slipSheetShift16	16 번 공정 간지 시프트 변수

3.2. 전체 순서도

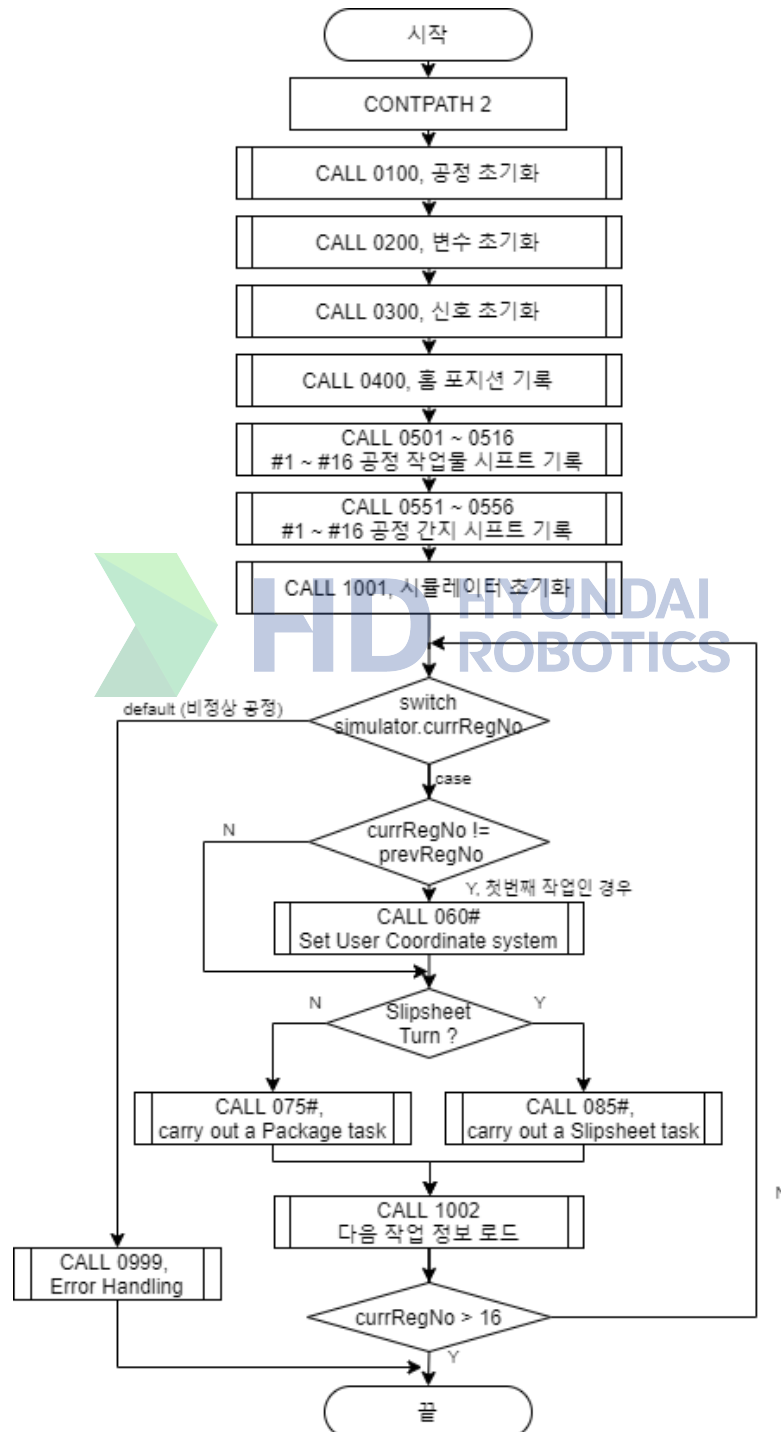


그림 3.1 표준 프로그램 전체 순서도

3.3. 프로그램 구성

표 3-5 표준 프로그램 JOB Number 및 역할

Job Number	역할	설명
1	메인	전체 흐름 관리
50	Vacuum On	그리퍼로 물건을 잡는 신호를 기록
51	Vacuum Off	그리퍼로 물건을 놓는 신호를 기록
100	공정 파라미터 설정	
200	각종 변수 초기화	
300	각종 신호 초기화	
400	홈пози션, 작업물 위치 포즈 프로그램	정의되어 있는 포즈 변수에 포즈 상수를 사용하여 위치를 기록해야 함
451~466	툴 제어 프로그램	각종 툴들의 위치를 제어
501~516	팔레트별 적재패턴 쉬프트 변수 프로그램	HRpal, HRpalware 에 의해 자동으로 생성됨
551~566	간지 쉬프트 변수 프로그램	HRpal, HRpalware 에 의해 자동으로 생성됨
601~616	팔레트 좌표계 생성 프로그램	팔레트에 사용자 좌표계를 생성해야 함 작업 원점, 해당 지점에서의 X 축, X-Y 평면상 점 기록 필요
701~716	팔레트 파라미터 설정	
751~766	팔레타이징 모션 프로그램	로봇을 움직여 팔레타이징 작업 수행
801~816	간지 파라미터 설정	
851~866	간지 모션 프로그램	로봇을 움직여서 간지를 얹어 놓음
999	에러 핸들링 프로그램	각종 에러 발생시, 표시해주기 위한 프로그램
1001	(시뮬레이션용) 테스트 셋업 프로그램	사용자가 설정한 값이 정상동작 하는지 확인을 위해 모의 값을 이용하게 해주는 부분
1002	(시뮬레이션용) 테스트 종료조건 체크	사이클 타임들의 체크를 위해 종료 조건을 기록해놓는 부분

3.3.1. 메인 프로그램

메인 프로그램은 아래와 같이 4 가지 부분으로 나눌 수 있습니다.

- (1) 각종 공정, 변수 등 초기화
- (2) 톨제어 및 시프트 변수 기록
- (3) 팔레타이징 조건 판단 및 수행
- (4) 에러 처리

표준 프로그램을 사용하여 공정을 운용하고자 할 때에는 반드시 한번은 메인 프로그램을 구동해야 합니다. 이는 메인 프로그램에서 각종 신호처리 및 초기화가 이루어 지기 때문인데, [Program 1 메인 프로그램 구성]에서 16 번째 라인 까지가 기본적인 초기화 부분 입니다. 메인 프로그램의 초기화 부분이 한번이라도 구동되어 정보를 저장하고 있다면, 팔레트 이동이나 간지 이동 등의 JOB 프로그램은 별 다른 처리 없이 정상적으로 구동될 수 있습니다.

Program 0001_MainJob.job

```

1 Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2   # Main Job Program
3   #-----
4   contpath 2
5   #-----
6   call 100 # process Initialize
7   call 200 # variable Initialize
8   call 300 # signal Initialize
9   call 400 # home position Initialize
10  #-----
11  call 451 # Record Tool Control for 1 Process
12  call 501 # Record Pallet Shift Information for 1 Process
13  call 551 # Record InsertSlipsheet Shift Information for 1 Process
14  call 452 # Record Tool Control for 2 Process
15  call 502 # Record Pallet Shift Information for 2 Process
16  #-----
17  call 1001 # Simulator Initialize
18  var startTimeStamp = timer()
19  #-----
20  *procStart
21  switch simulator.currRegNo
22
23  #-----
24  case 1 # Work # 1
25      if simulator.currRegNo != simulator.prevRegNo
26          call 601 # set User CRD
27      endif
28      if simulator.currPackageType == 7
29          call 851 # move Slipsheet (#1 process)
30      else

```

```

31         call 751 # move package (#1 process)
32     endif
33     simulator.prevRegNo = simulator.currRegNo
34     break
35
36     #-----
37     case 2 # Work # 2
38         if simulator.currRegNo != simulator.prevRegNo
39             call 602 # set User CRD
40         endif
41         call 752 # move package (#2 process)
42         simulator.prevRegNo = simulator.currRegNo
43         break
44     default
45         simulator.error = 1
46         call 999 # Error Handling
47         goto *procEnd
48         break
49     end_switch
50
51     #-----
52     call 1002 # load next work info
53     if simulator.currRegNo > 16 then *procEnd
54     goto *procStart # Next Process Start
55
56     #-----
57 *procEnd
58     print "Total Cycle Time =", timer()-startTimeStamp
59     end

```

3.3.2. Vacuum On, Off

Vacuum On, Off 프로그램(50, 51 번 JOB 프로그램)은 팔레트 이동 프로그램 내에서 CALL 하는 형태로 사용됩니다. 기본적으로 생성되는 JOB 프로그램에는 fb0.do10 을 신호로 사용하여 정의되어 있으며, 실제 공정에 적용할 때에는 해당 공정에 맞는 신호로 JOB 프로그램을 수정하면 됩니다.

3.3.3. 공정 파라미터 설정

공정 파라미터 설정(100 번 JOB 프로그램)은 해당 공정의 사용 유/무를 결정하기 위한 JOB 프로그램 입니다. processInfo[0] ~ processInfo[15] 의 값에 공정 관련 정보가 작성되어 있고, 특정 공정을 Disable 로 설정하려면 공정 등록 정보를 주석 처리하면 됩니다.

할당된 공정 파라미터 값은 향후 모니터링을 통해 해당 공정의 활성화 여부를 확인하기 위해 사용할 수 있습니다. 대부분의 값은 HRpal 에서 자동으로 생성됩니다. 그러나 아래 샘플 코드의 19 번, 35 번 라인의 pickupHeight 값은 필요 시 사용자가 직접 수정하여 사용 가능합니다. 기본값은 작업물 높이의 1.5 배 값입니다.(그림 3-2)

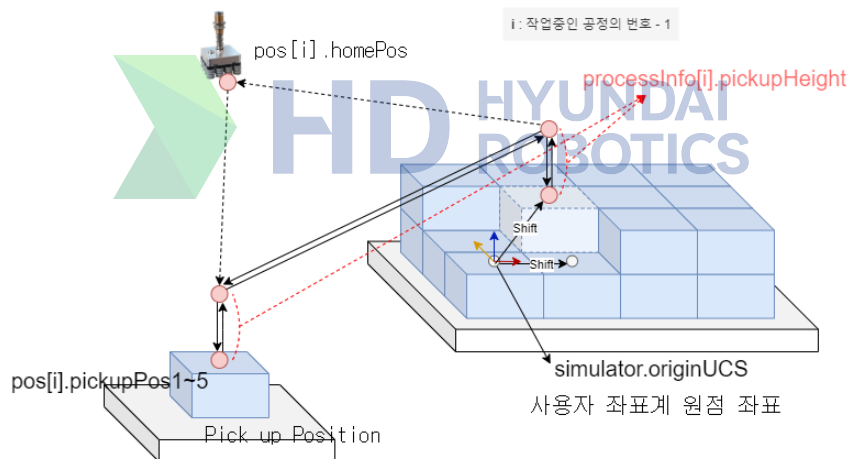


그림 3.2 팔레타이징 작업 시 사용되는 좌표 정보

Program 0100_ProcessSetting.job

```

1 Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2   # Process Parameter Setting
3   global processInfo = Array(16)
4
5   #-----
6   # Process Info for #1 Process
7   processInfo[0] = {regNo:0, totalPackages:0, oddPackages:0, evenPackages:0,
8                     topPackages:0, totalSlipsheet:0, firstLayerSlipSheet:0,
9                     topLayerSlipSheet:0, packageLength:0, packageWidth:0,
10                    packageHeight:0, pickupHeight:0}
11
12   processInfo[0].regNo = 1 # RegisterNo
13   processInfo[0].totalPackages = 24 # TotalPackageCount

```

```

10    processInfo[0].oddPackages = 12 # OddPackageCount
11    processInfo[0].evenPackages = 12 # EvenPackageCount
12    processInfo[0].topPackages = 12 # TopPackageCount
13    processInfo[0].totalSlipsheet = 1 # SlipsheetStackLayer
14    processInfo[0].firstLayerSlipSheet = false # SlipsheetStackFirst
15    processInfo[0].topLayerSlipSheet = false # SlipsheetStackTop
16    processInfo[0].packageLength = 180 # Package Length
17    processInfo[0].packageWidth = 450 # Package Width
18    processInfo[0].packageHeight = 160 # Package Height
19    processInfo[0].pickupHeight = 240 # PickupHeight
20
21    #-----
22    # Process Info for #2 Process
23    processInfo[1] = {regNo:0, totalPackages:0, oddPackages:0, evenPackages:0,
                       topPackages:0, totalSlipsheet:0, firstLayerSlipSheet:0,
                       topLayerSlipSheet:0, packageLength:0, packageWidth:0,
                       packageHeight:0, pickupHeight:0}
24    processInfo[1].regNo = 2 # RegisterNo
25    processInfo[1].totalPackages = 36 # TotalPackageCount
26    processInfo[1].oddPackages = 18 # OddPackageCount
27    processInfo[1].evenPackages = 18 # EvenPackageCount
28    processInfo[1].topPackages = 18 # TopPackageCount
29    processInfo[1].totalSlipsheet = 0 # SlipsheetStackLayer
30    processInfo[1].firstLayerSlipSheet = false # SlipsheetStackFirst
31    processInfo[1].topLayerSlipSheet = false # SlipsheetStackTop
32    processInfo[1].packageLength = 200.52 # Package Length
33    processInfo[1].packageWidth = 300.11 # Package Width
34    processInfo[1].packageHeight = 200.2 # Package Height
35    processInfo[1].pickupHeight = 300.3 # PickupHeight
36    end

```

3.3.4. 각종 변수, 신호 초기화

변수 초기화 프로그램(200 번 JOB 프로그램)은 표준 프로그램에서 시뮬레이션 도중 사용되는 변수값들의 초기값을 지정해주는 프로그램 입니다.

Program 0200_VariableInit.job

```

1  Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2      # Initialize Variable
3      global simulator = {currRegNo:0, prevRegNo:0, currActionNo:0, currPackageType:0,
                           prevPackageType:0, cntPackage:0, cntSlipsheet:0, cntLayer:0,
                           originUCS:0, error:0, homePos:0, nearPickupPos:0,
                           pickupPos:0, nearTargetPos:0, targetPos:0}
4      print "각종 변수값을 수정 후, PRINT, STOP 은 삭제 하세요."
5      stop
6      simulator.currRegNo = 0
7      simulator.prevRegNo = 0
8      simulator.currPackageType = 1
9      simulator.prevPackageType = 1
10     simulator.cntPackage = 0 # Package Counter Reset
11     simulator.cntSlipsheet = 0 # SlipSheet Counter Reset
12     simulator.cntLayer = 0 # Layer Counter Reset
13     end

```

신호 초기화 프로그램(300 번 JOB 프로그램)은 해당 공정 수행 전에 초기화해야 하는 신호들을 기록해놓기 위해 사용됩니다. 초기에는 아무런 내용이 정의되어 있지 않으며, 공정에 맞게 신호를 사전에 기록해놓습니다.

3.3.5. 홈포지션, 작업물 위치 포즈 프로그램

홈포지션, 작업물 위치 포즈 프로그램(400 번 JOB 프로그램)은 특정 공정의 홈포지션의 위치와 작업물을 픽업하기 위해 이동해야 하는 위치를 기록하는 프로그램입니다. homePos 변수는 Home Position 으로 사용되고, pickupPos1 ~ pickupPos5 변수는 각각의 공정에 대한 Pickup Position 을 지정하는데 사용합니다.(그림 3-2) 하나의 공정은 최대 5 개의 다른 작업물을 대상으로 공정을 수행할 수 있습니다.

Program 0400_HomePos.job

```

1 Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2   # Home, Package, Slipsheet Position
3   global pos = Array(16)
4   print "각종 변수값을 수정 후, PRINT, STOP 은 삭제 하세요."
5   stop
6
7   #-----
8   # Position Info for #1 Process
9   pos[0] = {homePos:0, pickupPos1:0, pickupPos2:0, pickupPos3:0, pickupPos4:0,
10             pickupPos5:0, SlipSheetPos:0}
11   pos[0].homePos = cpo("robot", "cur") # Home Position #1
12   pos[0].pickupPos1 = cpo("robot", "cur") # Pickup Position #Work 1_1
13   pos[0].pickupPos2 = cpo("robot", "cur") # Pickup Position #Work 1_2
14   pos[0].pickupPos3 = cpo("robot", "cur") # Pickup Position #Work 1_3
15   pos[0].pickupPos4 = cpo("robot", "cur") # Pickup Position #Work 1_4
16   pos[0].pickupPos5 = cpo("robot", "cur") # Pickup Position #Work 1_5
17   pos[0].SlipSheetPos = cpo("robot", "cur") # Slipsheet Position #1
18   #-----
19   # Position Info for #2 Process
20   pos[1] = {homePos:0, pickupPos1:0, pickupPos2:0, pickupPos3:0, pickupPos4:0,
21             pickupPos5:0, SlipSheetPos:0}
22   pos[1].homePos = cpo("robot", "cur") # Home Position #2
23   pos[1].pickupPos1 = cpo("robot", "cur") # Pickup Position #Work 2_1
24   pos[1].pickupPos2 = cpo("robot", "cur") # Pickup Position #Work 2_2
25   pos[1].pickupPos3 = cpo("robot", "cur") # Pickup Position #Work 2_3
26   pos[1].pickupPos4 = cpo("robot", "cur") # Pickup Position #Work 2_4
27   pos[1].pickupPos5 = cpo("robot", "cur") # Pickup Position #Work 2_5
28   end

```

3.3.6. 톨 제어 프로그램

톨 제어 프로그램(451 번~466 번 JOB 프로그램)은 각종 지그나 위치를 조절하는데 사용되는 톨들을 제어하는 부분을 정의하기 위해 사용합니다. 톨 제어 프로그램은 공정에 따라서 사용 유/무가 다르고, 사용이 필요한 위치도 다르기 때문에 별도로 호출 위치가 지정되어 있지 않습니다. 그러므로 공정에 맞도록 해당 프로그램에 사용자가 제어 프로그램을 작성한 후, 호출 위치를 직접 지정하여 사용해야 합니다.

3.3.7. 팔레트별 적재패턴, 간지 시프트 변수 프로그램

팔레트별 적재 패턴(501~516 번 JOB 프로그램)과 간지 시프트 변수 프로그램(551~566 번 JOB 프로그램)은 적재 패턴과 간지의 시프트 값들이 저장되어 있는 프로그램입니다. 이는 HRpal 을 사용하여 만든 패턴이 자동으로 저장되며, 사용자가 직접 시프트 값을 수정할 수 있습니다.

그리고 각 시프트 변수에 주석으로 자동으로 표기되는 것은 층_순서의 방식으로 표기되며, 예를 들면 1_3 은 1 층의 3 번째 작업물을 의미합니다.

Program 0501_PatternShift_1.job

```

1 Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2
3 global packageShift1 = Array(24) # init packageShift Variable
4 packageShift1[0] = Shift(0.000, 0.000, 0.000, 0.000, 0.000, 0.000, "u1") # 1_1
5 packageShift1[1] = Shift(0.000, 180.000, 0.000, 0.000, 0.000, 0.000, "u1") # 1_2
6 packageShift1[2] = Shift(0.000, 360.000, 0.000, 0.000, 0.000, 0.000, "u1") # 1_3
7 packageShift1[3] = Shift(-315.000, 135.000, 0.000, 0.000, 0.000, 90.000, "u1") # 1_4
8 packageShift1[4] = Shift(135.000, 675.000, 0.000, 0.000, 0.000, 90.000, "u1") # 1_5
9 packageShift1[5] = Shift(-495.000, 135.000, 0.000, 0.000, 0.000, 90.000, "u1") # 1_6
10 packageShift1[6] = Shift(-45.000, 675.000, 0.000, 0.000, 0.000, 90.000, "u1") # 1_7
11 packageShift1[7] = Shift(-675.000, 135.000, 0.000, 0.000, 0.000, 90.000, "u1") # 1_8
12 packageShift1[8] = Shift(-225.000, 675.000, 0.000, 0.000, 0.000, 90.000, "u1") # 1_9
13 packageShift1[9] = Shift(-540.000, 450.000, 0.000, 0.000, 0.000, 0.000, "u1") # 1_10
14 packageShift1[10] = Shift(-540.000, 630.000, 0.000, 0.000, 0.000, 0.000, "u1") # 1_11
15 packageShift1[11] = Shift(-540.000, 810.000, 0.000, 0.000, 0.000, 0.000, "u1") # 1_12
16 packageShift1[12] = Shift(0.000, 0.000, 170.000, 0.000, 0.000, 0.000, "u1") # 2_1
17 packageShift1[13] = Shift(0.000, 180.000, 170.000, 0.000, 0.000, 0.000, "u1") # 2_2
18 packageShift1[14] = Shift(0.000, 360.000, 170.000, 0.000, 0.000, 0.000, "u1") # 2_3
19 packageShift1[15] = Shift(-315.000, 135.000, 170.000, 0.000, 0.000, 90.000, "u1") # 2_4
20 packageShift1[16] = Shift(135.000, 675.000, 170.000, 0.000, 0.000, 90.000, "u1") # 2_5
21 packageShift1[17] = Shift(-495.000, 135.000, 170.000, 0.000, 0.000, 90.000, "u1") # 2_6
22 packageShift1[18] = Shift(-45.000, 675.000, 170.000, 0.000, 0.000, 90.000, "u1") # 2_7
23 packageShift1[19] = Shift(-675.000, 135.000, 170.000, 0.000, 0.000, 90.000, "u1") # 2_8
24 packageShift1[20] = Shift(-225.000, 675.000, 170.000, 0.000, 0.000, 90.000, "u1") # 2_9
25 packageShift1[21] = Shift(-540.000, 450.000, 170.000, 0.000, 0.000, 0.000, "u1") # 2_10
26 packageShift1[22] = Shift(-540.000, 630.000, 170.000, 0.000, 0.000, 0.000, "u1") # 2_11
27 packageShift1[23] = Shift(-540.000, 810.000, 170.000, 0.000, 0.000, 0.000, "u1") # 2_12
28 end

```

3.3.8. 팔레트 좌표계 생성 프로그램

HRpal 에서 생성된 패턴의 시프트 데이터는 모두 1 번 적재물의 위치에 대한 상대적인 시프트 값으로 기록되어 있습니다.(그림 3-2) 그러므로 팔레트 상에서 최초 1 번 위치(혹은 티칭 위치, *로 표기됨)가 적재될 위치를 기준으로 하여 사용자 좌표계를 만드는 작업이 필요하며, 이는 팔레트 좌표계 생성 프로그램(601~616 번 JOB 프로그램) 파일에 정의되어 있습니다.

사용자 좌표계는 3 개의 좌표값을 기록하는 것으로 생성할 수 있으며, 기록해야 할 좌표값은 아래와 같습니다.(그림 3-3)

- P1 - 가장 최초에 적재를 시작할 적재물의 위치를 기록합니다. 이는 동시에 사용자 좌표계상의 원점입니다.
- P2 - P1 위치를 기준으로 X 축의 한 점을 임의로 설정합니다. (+ 방향 X 축)
- P3 - P1 위치를 기준으로 X-Y 평면 위의 한 점을 임의로 설정합니다. (+ 방향 X-Y 평면)

사용자 좌표계는 오른손 좌표계를 기준으로 생성되기 때문에, 적재 시작 위치(오른쪽 위, 왼쪽 아래 등)에 따라 P1, P2, P3 점의 기록 위치가 팔레트 외부 좌표일 수 있습니다. (그림 3-3 참고)

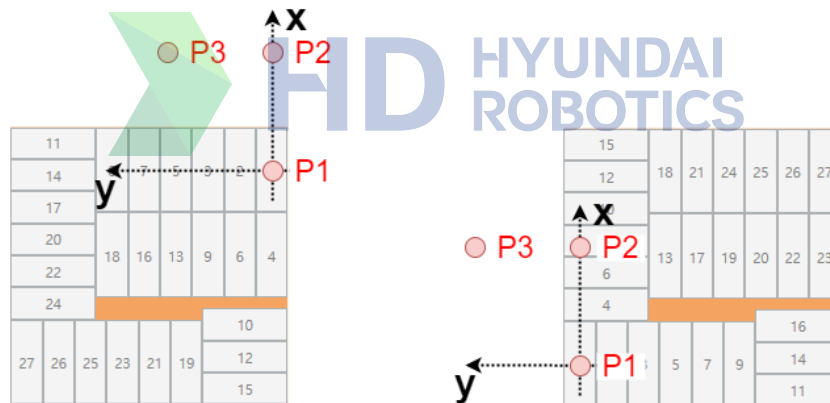


그림 3.3 사용자 좌표계 기록 예시

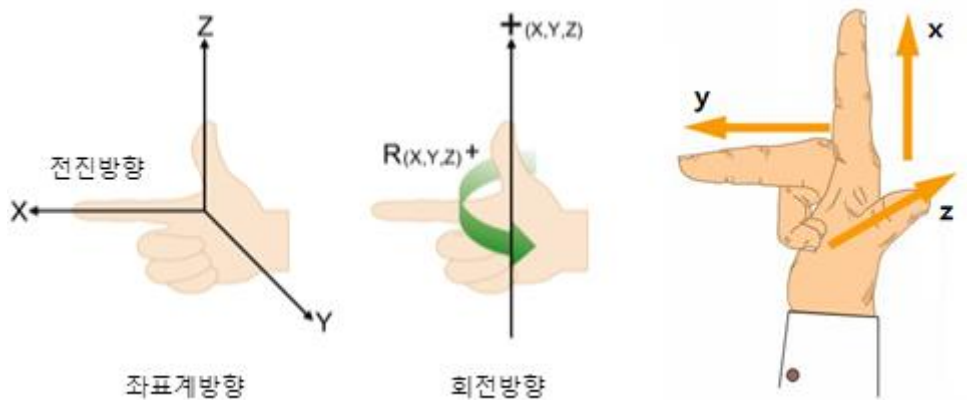


그림 3.4 오른손 좌표계의 구성 형태

Program 0601_UserCrd_1.job

```
1 Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2 # HHI User Coordinate Define Program
3 print "사용자 좌표계를 포즈상수값을 사용하여 기록하세요.
   완료 후 PRINT, STOP 은 삭제 하세요."
4 stop
5 var p1 = cpo("robot", "cur") # Origin Position
6 var p2 = cpo("robot", "cur") # Any position on the +X-axis
7 var p3 = cpo("robot", "cur") # Any position on the X-Y plane
8 var result = mkucs(1, p1, p2, p3) # Register user coordinate system
9 selucrd 1
10 simulator.originUCS = p1
11 end
```



3.3.9. 팔레트, 간지 파라미터 설정

팔레트, 간지 파라미터 설정 프로그램(701~716 번 JOB 프로그램)은 팔레타이징의 진행을 위해 각종 시프트 변수를 대입하거나 팔레타이징에 필요한 변수들을 초기화 하는 등의 일을 수행합니다. 기본적으로는 해당 JOB 프로그램의 변수를 수정할 부분이 없지만, 경우 Step 을 추가하거나, 변경하고자 하는 경우에는 해당 JOB 프로그램을 수정해서 프로그램을 작성할 수 있습니다.

Program 0701_PalParamSet_1.job

```

1 Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2 # Palletize Parameter Setting #1
3 simulator.error = 0 # Error Handling Variable Init
4 if simulator.cntPackage == processInfo[0].totalPackages
5     simulator.cntPackage = 0
6     simulator.cntSlipsheet = 0
7     simulator.cntLayer = 0
8 endif
9
10 var shiftHeight = Shift(0, 0, processInfo[0].pickupHeight, 0, 0, 0, "robot")
11                                     # shift position for lift or place down
12 var shiftPackage = packageShift1[simulator.cntPackage]
13
14 var pickupPos
15 switch simulator.currPackageType
16 case 1
17     pickupPos = pos[0].pickupPos1
18     break
19 case 2
20     pickupPos = pos[0].pickupPos2
21     break
22 case 3
23     pickupPos = pos[0].pickupPos3
24     break
25 case 4
26     pickupPos = pos[0].pickupPos4
27     break
28 case 5
29     pickupPos = pos[0].pickupPos5
30     break
31 default
32     pickupPos = pos[0].pickupPos1
33     break
34 end_switch
35
36 simulator.homePos = pos[0].homePos # Home position during palletizing
37 simulator.nearPickupPos = pickupPos+shiftHeight

```

```

37         # Just above the position to pick up the package or slipsheet.
38     simulator.pickupPos = pickupPos # The position to pick up the package or slipsheet.
39     simulator.nearTargetPos = simulator.originUCS+shiftPackage+shiftHeight
40         # just above the position to put down the package or slipsheet.
41     simulator.targetPos = simulator.originUCS+shiftPackage
42         # The position to put down the package or slipsheet.
43     end
```



3.3.10. 팔레타이징, 간지 모션 프로그램

팔레타이징, 간지 모션 프로그램은 실제 로봇을 움직이기 위한 MOVE 문들의 조합으로 구성되어 있습니다. 표준 프로그램은 연속적인 동작을 위해 4 개의 위치값(Package, Pick up, Near place down, place down)을 사용합니다. 이들 4 개의 값은 앞서 4.3.5 홈포지션, 작업을 위치 프로그램, 4.3.8 팔레트 좌표계 생성 프로그램과 4.3.4 각종 변수, 신호 초기화에서 지정한 들어올릴 높이 값을 사용하여 정의합니다. 그리고 Home Position은 다른 작업물이 진입했거나, 간지 삽입을 하러 가야하는 등 연속적이지 않은 동작을 수행해야 될 경우에 이동할 초기 위치를 의미합니다.

만일, S4. Up 과 S5. Near Place Down 사이에 다른 경유점을 추가하고자 할 때에는 S4 와 S5 사이에 새로운 MOVE문을 추가하여 사용하면 됩니다.

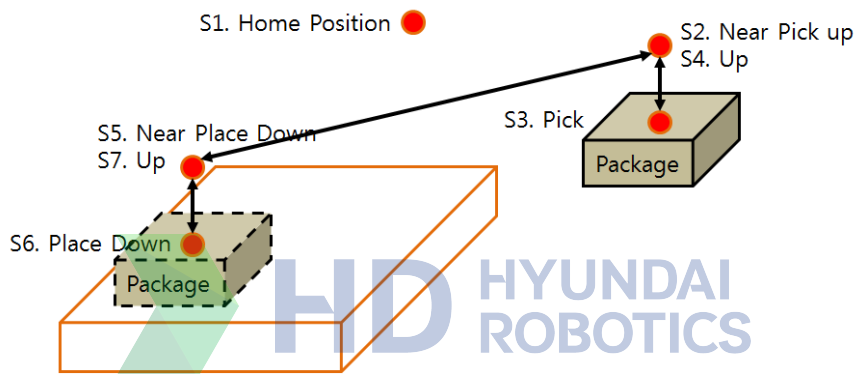


그림 3.5 팔레타이징 동작 위치값

Program 0751_PalJob_1.job

```

1  Hyundai Robot Job File: { version: 1.6, mech_type: "0()", total_axis: 0, aux_axis: 0 }
2      # Palletize Job Program #1
3      call 701 # Motion Param Set
4      if (simulator.currRegNo!=simulator.prevRegNo or
          simulator.currPackageType!=simulator.prevPackageType)
5  S1      move P,tg=simulator.homePos,spd=90%,accu=1,tool=0 # Home Position
6  endif
7  S2      move P,tg=simulator.nearPickupPos,spd=90%,accu=5,tool=0 # Near Pick up
8  S3      move L,tg=simulator.pickupPos,spd=90%,accu=0,tool=0 # Pick
9          call 0050 # Vacuum On
10 S4      move L,tg=simulator.nearPickupPos,spd=90%,accu=5,tool=0 # Up
11 S5      move P,tg=simulator.nearTargetPos,spd=90%,accu=5,tool=0 # Near Place Down
12 S6      move L,tg=simulator.targetPos,spd=90%,accu=0,tool=0 # Place Down
13          call 0051 # Vacuum off
14          simulator.cntPackage = simulator.cntPackage+1 # Counter increment
15          simulator.prevPackageType = simulator.currPackageType
16 S7      move L,tg=simulator.nearTargetPos,spd=90%,accu=5,tool=0 # Up
17      end
    
```



3.4. 표준 프로그램의 기본적인 사용

팔레타이징 표준 프로그램의 대부분은 자동으로 구성되고, 사용자가 일부 부분만 수정하면 동작하는 방식으로 구성되어 있습니다. 설정해야 하는 부분으로는 변수 설정과 위치 기록의 두 가지 부분이 있고, 해당 부분만 수정하면 로봇이 자동으로 움직이도록 되어 있기 때문에 매우 빠르게 로봇의 움직임을 체크해볼 수 있습니다.

3.4.1. 변수 설정

자동으로 생성된 표준 프로그램에서 사용자가 변경해야 하는 변수는 processInfo 배열의 pickupHeight 값입니다. 해당 변수는 작업물을 들어 올리거나 내려 놓을 때, 얼마나 높이 들어올릴 지 혹은 내려 놓을지를 지정하는 변수입니다.

그 외의 나머지 변수는 필요에 따라 자유롭게 변경할 수 있고, 자세한 내용은 [3.1. 변수 구성]을 참고하시면 됩니다.

3.4.2. 위치 기록

표준 프로그램에서 기록해야 할 기본 위치는 크게 2가지가 있습니다. 첫 번째는 적재물의 위치를 나타내는 포즈 변수이고, 두 번째는 팔레트의 상의 사용자 좌표계를 만들기 위한 변수입니다.

(1) 픽업 위치와 관계된 포즈 기록

적재물 위치와 관계된 변수는 0400_HomePos.job 파일에 정의되어 있고, 작업물 픽업 위치, 간지 픽업 위치 그리고 홈 포지션 위치가 각 공정 별로 정의되어 있습니다. 사용자는 실제 로봇 혹은 시뮬레이션에서 티치펜던트를 통해 해당 위치로 로봇을 이동 시킨 후, 해당 포즈 변수를 포즈 상수 값으로 변경시키면 됩니다. ([3.3.5. 홈포지션, 작업물 위치 포즈 프로그램] 참고)

하나의 적재 패턴에 대해서 한 개의 적재물만 사용되는 경우에는 Home Position 과 Pickup Position #1 만을 티칭하면 되고, 여러 개의 적재물이 사용된다면, 해당 되는 Pickup Position 을 모두 티칭하면 됩니다.

(2) 팔레트 상에 사용자 좌표계 정의

HRpal 혹은 HRpalware 에서 생성된 패턴의 시프트 데이터는 모두 1 번 적재물의 위치에 대한 상대적인 시프트 값으로 기록되어 있습니다. 그러므로 팔레트 상에서 최초 1 번 위치(혹은 티칭 위치, *로 표기됨)가 적재될 위치를 기준으로 하여 사용자 좌표계를 만드는 작업이 필요하며, 이는 팔레트 좌표계 생성 프로그램(601~616 번 JOB 프로그램) 파일에 정의되어 있습니다.

자세한 내용은 [3.3.8 팔레트 좌표계 생성 프로그램]을 참고하시기 바랍니다.



GRC: 경기도 성남시 분당구 분당수서로 477

대구: 대구광역시 달성군 유가읍 테크노순환로 3 길 50

울산: 울산광역시 북구 매곡산업로 21 자동차조선기술관 201-5 호

중부: 충남 아산시 염치읍 송곡길 161

광주: 광주광역시 광산구 평동산단로 170-3 B 동 101 호

ARS 1588-9997 | 1 로봇영업 2 서비스영업 3 구매상담 4 고객지원 5 투자문의 6 채용 및 일반 문의

www.hyundai-robotics.com