**공정 트윈 실습 프로젝트**

**시나리오에 따른 공정 시뮬레이션**

1조 – 안형석, 박민준, 오영택, 김건호

**목차**

1. **프로젝트 개요**
2. **프로젝트 팀 소개**
3. **프로젝트 진행 과정**
4. **프로젝트 개요**

|  |  |
| --- | --- |
| 프로젝트명 | 시나리오에 따른 공정 시뮬레이션 |
| 프로젝트 기간 | 2023/02/02 ~ 2023/03/28 |
| 목적 | 구조에 따른 생산량의 변화를 통해 가장 효율적인 구조를 파악 |
| 핵심 기술 | 디지털트윈 플랫폼인 WAISER 엔진을 사용하며, 데이터베이스와 유니티를 이용한 시뮬레이션 결과의 시각화 |
| 기대 효과 | 공정 구조에 따른 생산량의 변화를 봄으로써 구조를 설계하기 전 시뮬레이션을 통해 어떤 구조가 효율적으로 제품생산을 하는지 파악할 수 있음 |

1. **프로젝트 팀 소개**

|  |  |
| --- | --- |
| 이 름 | 역 할 |
| 안형석 | 팀장, 와이저를 활용한 디지털트윈 및 전반적인 조율 |
| 김건호 | 유니티를 활용한 시각화 담당. 회의록 작성 |
| 박민준 | 와이저를 활용한 디지털트윈 |
| 오영택 | 데이터베이스를 활용해 결과값 분석 및 시각화 |

1. **프로젝트 진행 과정**
2. 시나리오 3개 설계
3. 공장의 로직 설계
4. 상태천이도 제작
5. WAiSER를 사용해 상태천이도 대로 공장 모델링
6. DB 설계 및 시각화(ERD)
7. DB와 모델 연동
8. DB와 Unity 연동
9. Unity 작동파일 확정 및 작동 영상화
10. 결과 데이터 분석
11. 문서 작성(결과 보고서 등)
12. **프로젝트 목적**

각각 다른 설비와 구조를 가진 시나리오 3개를 만든 후, 시나리오 별로 생산량/가동률이 어떻게 변화되는지 파악해서 생산량/가동률이 가장 높은 구조(가장 효율적인 구조)가 어느 구조인지 파악하기 위함. 이로 인해 공정 구조를 효율적으로 설계 및 제작이 가능함.

1. **프로젝트 달력**

** **

1. **데이터 흐름도**

도표이(가) 표시된 사진

자동 생성된 설명

유저가 DB에 직접 근간이 되는 데이터 project\_list(프로젝트 목록), object\_list(설비 목록), obj\_coup\_list(설비 연결 상태)를 입력하면, 와이저에서 DB의 해당 시나리오 project\_list, object\_list, obj\_coup\_list를 받아 시뮬레이션을 작동시키고, 그 결과물을 DB에 업로드 후, 유니티에서 DB데이터를 다운받아 시각화 한다.

1. **모델 구조**
2. **시나리오 설계**

* 시나리오 1.

차트이(가) 표시된 사진

자동 생성된 설명

기본적인 설비 구조. 제네레이터, 공정설비, 적재설비가 각각 하나씩 있는 일직선의 간단한 구조이다.

이 시나리오를 통해 기본적인 설비구조에서의 생산량/가동률을 알 수 있다.

* 시나리오 2.

차트이(가) 표시된 사진

자동 생성된 설명

시나리오 1과 달리 공정설비들은 2개의 트랙을 받으며 10번 설비는 2개의 적재설비 중 적재된 개수가 적은 하나로 제품을 보내는 구조이다. 이 시나리오를 통해서 통로들이 다대 다 대응인 공정 설비가 2개인 환경에서의 생산량/가동률을 알 수 있다.

* 시나리오 3

차트이(가) 표시된 사진

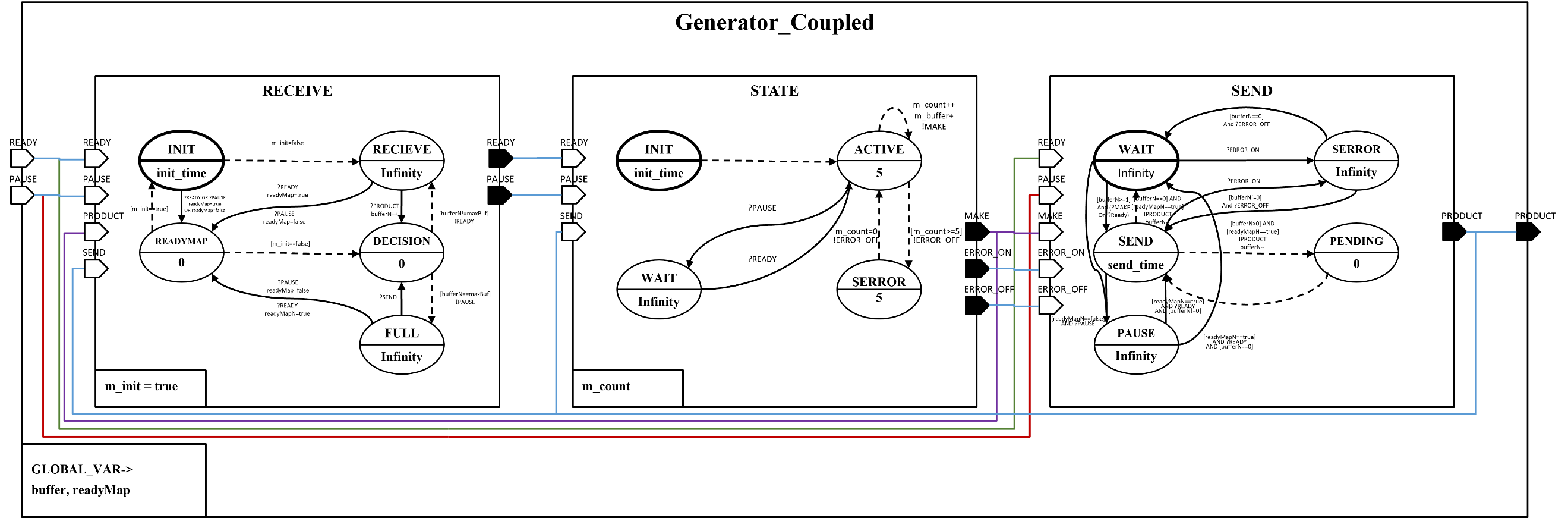
자동 생성된 설명

.시나리오 1, 2와 달리 설비들은 Generator이나 설비등과 연결된 여러 개의 트랙을 받고 일부 트랙은 둘 중 하나의 통로로 제품을 내보내는 구조이다. 이 시나리오를 통해서 통로들이 다대 다 대응인 공정 설비가 여러 개인 환경에서의 생산량/가동률을 알 수 있다.

1. **상태 천이도**

* **Generator**

제품 투입 설비. 제품을 투입해서 다음 설비로 보낸다. 5초마다 하나씩 제품을 투입. 다음 설비가 받을 수 없는 상태일 시 제품을 보내지 않음



* **Track**

트랙/이동 설비. 제품을 받아서 다음 설비로 보냄. 다음 설비가 받을 수 없는 상태일 시 제품을 보내지 않음

차트이(가) 표시된 사진

자동 생성된 설명

* **Process**

공정 설비. 각 공정설비나 Generator에서 제품을 받아서 새로운 제품을 생산하여 다음 설비로 보냄.

다음 설비가 받을 수 없는 상태일 시 제품을 보내지 않음

차트이(가) 표시된 사진

자동 생성된 설명

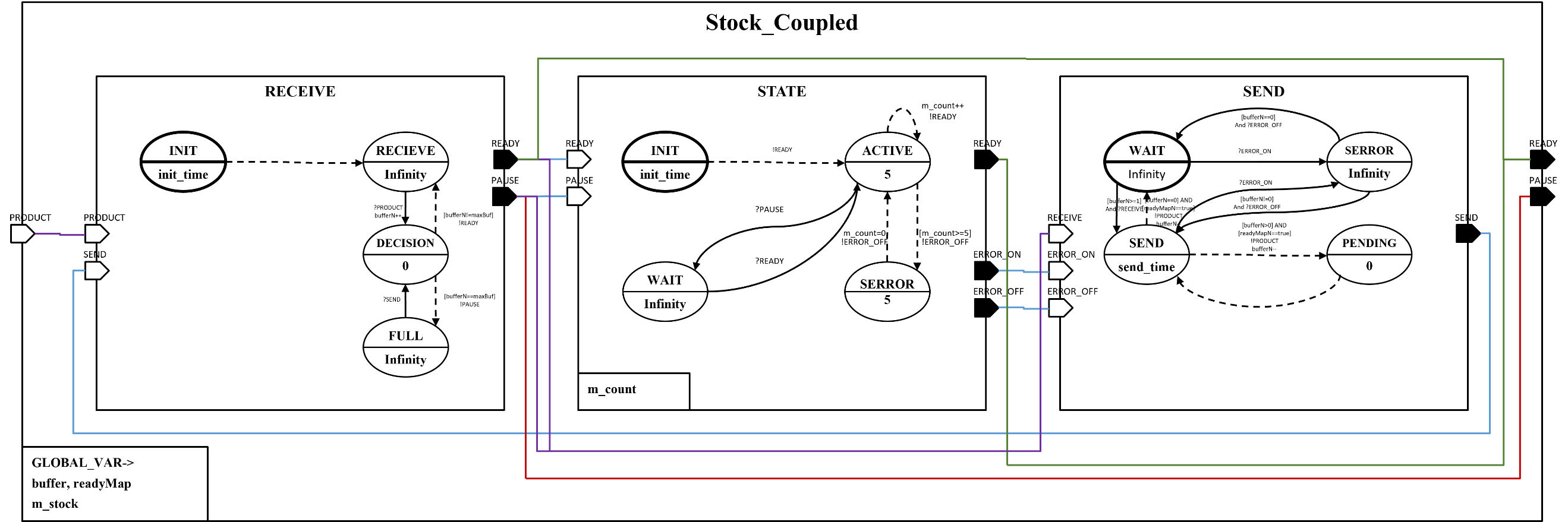
공정설비가 다수의 트랙을 받을 경우 다음과 같이 변경됨

차트이(가) 표시된 사진

자동 생성된 설명

* **STOCK**

적재 설비. 생산된 물건을 적재하는 설비. 얼만큼 이곳에서 적재되었는지를 통해 생산량을 알 수 있다.



* **GEN-TRACK/TRACK-PROC**

도표이(가) 표시된 사진

자동 생성된 설명

* **PROC-TRACK/TRACK-STOCK**

도표이(가) 표시된 사진

자동 생성된 설명

* **TRACK-TRACK**

텍스트이(가) 표시된 사진

자동 생성된 설명

1. **프로젝트 산출물**

* WAiSER 시뮬레이션 결과

텍스트, 전자제품이(가) 표시된 사진

자동 생성된 설명

* 1번 시나리오

텍스트, 전자제품이(가) 표시된 사진

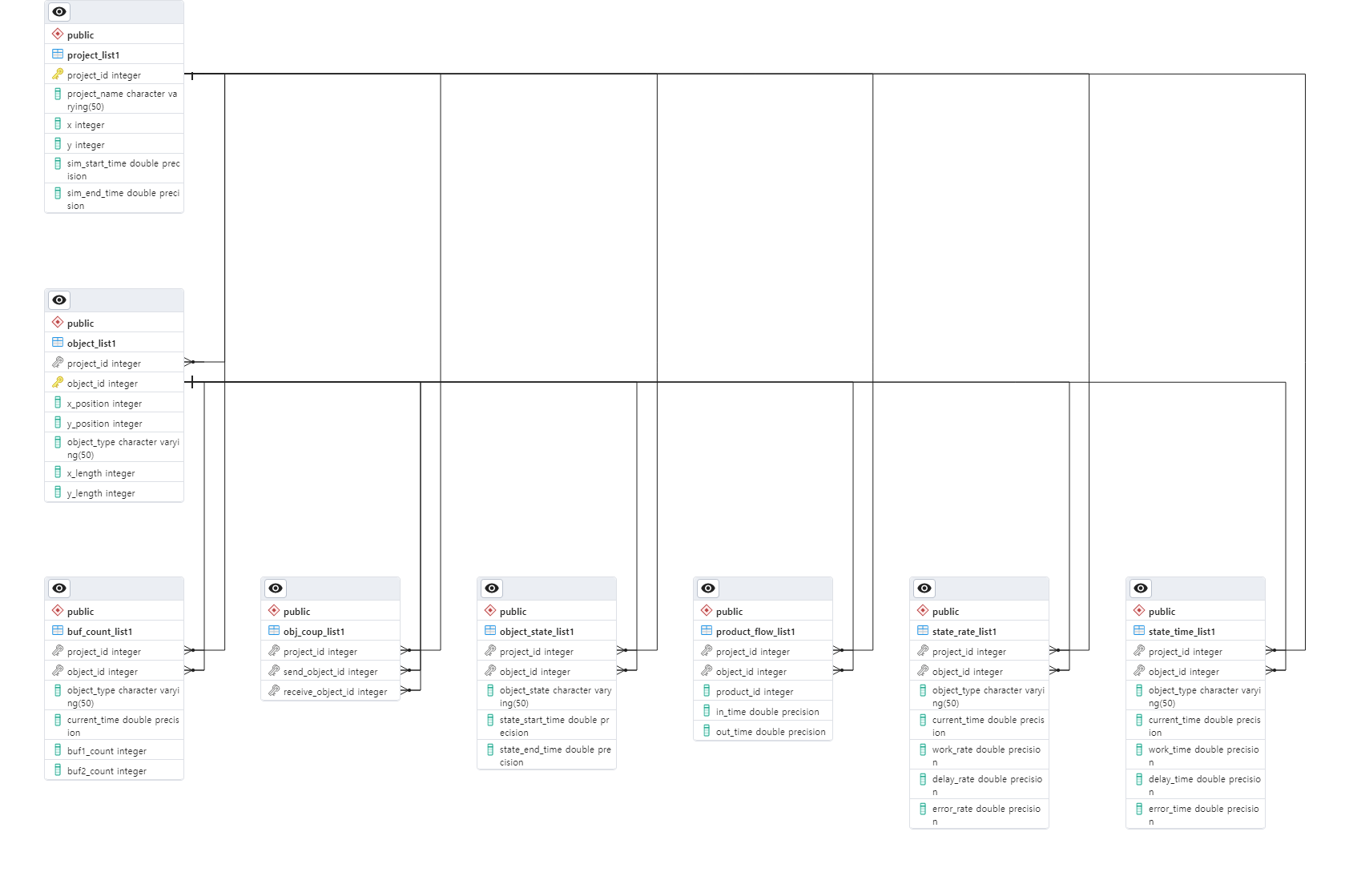
자동 생성된 설명

* 2번 시나리오

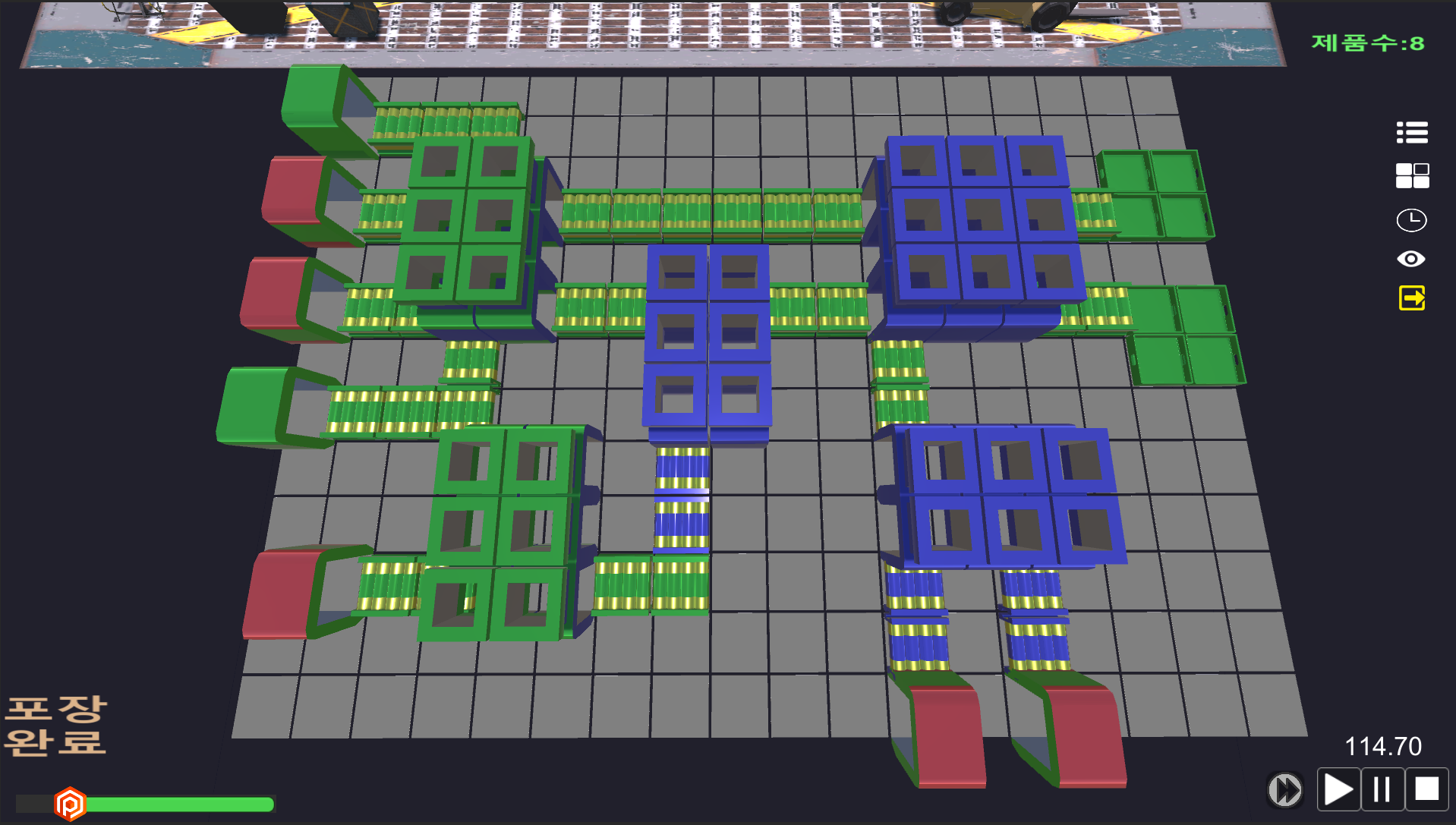
텍스트, 전자제품이(가) 표시된 사진

자동 생성된 설명

* 3번 시나리오
* DB



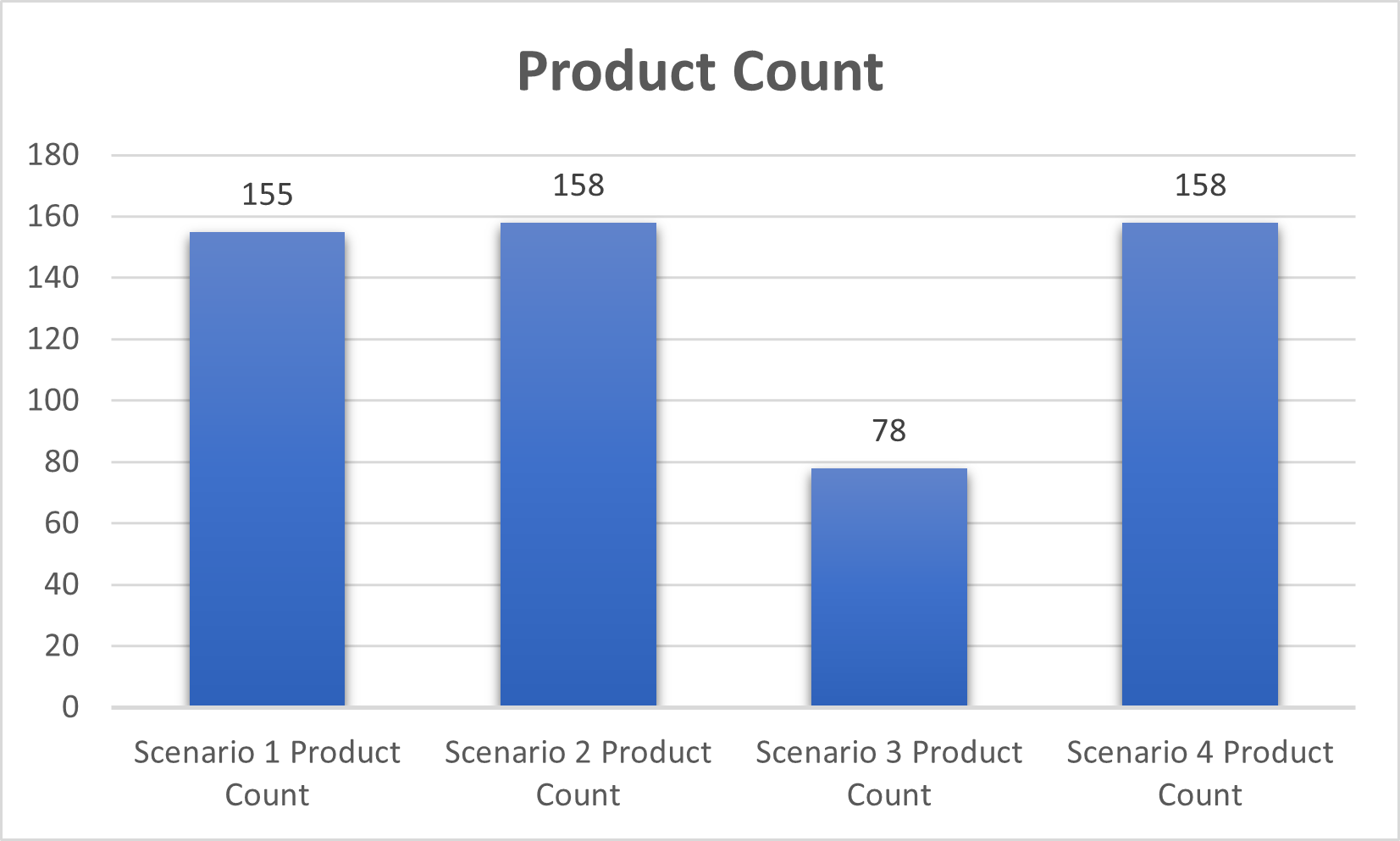
* ERD
* Unity



* 시나리오 3 시각화

1. **실험 결과**

* 생산량



단순한 구조를 가진 1번 시나리오가 가장 많은 제품을 생산할 것으로 예상하였으나, 2, 4번 시나리오가 가장 많은 제품을 생산하였다. 3번 시나리오는 가장 복잡한 구조를 가져 생산량이 가장 적을 것으로 예상하였고, 맞아 떨어졌다. 4번 시나리오는 1번 시나리오에 STOCK과 STOCK을 연결하는 트랙 하나를 추가한 구조인데, STOCK이 많을수록 생산량이 증가할 걸로 예상하였고, 예상대로 4번 시나리오가 1번 시나리오보다 생산량이 많았다.

* 각 시나리오별 설비 상태 비율(가동률)

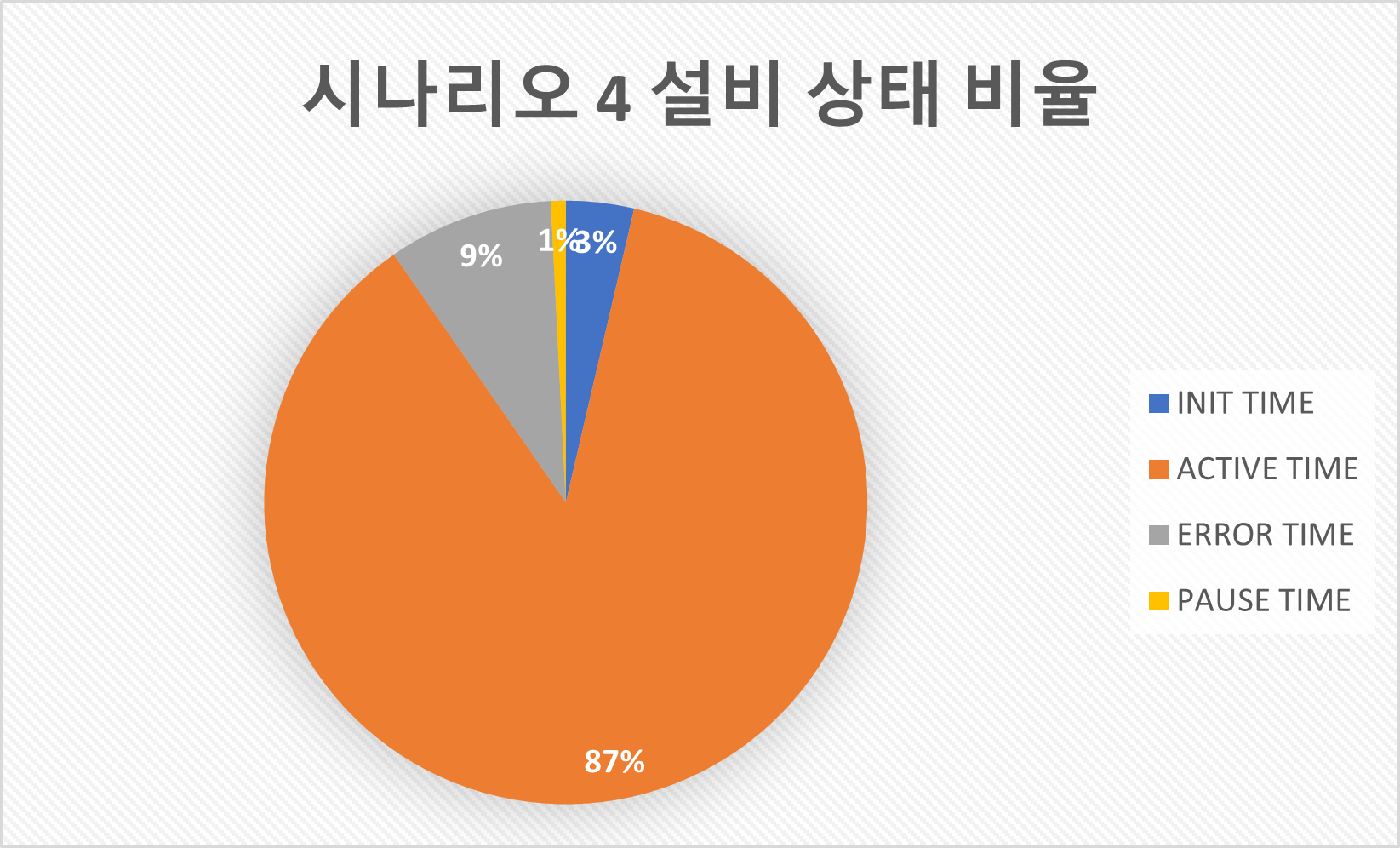
차트, 파이 차트이(가) 표시된 사진

자동 생성된 설명

차트, 파이 차트이(가) 표시된 사진

자동 생성된 설명차트, 파이 차트이(가) 표시된 사진

자동 생성된 설명



각 시나리오 별로 설비 구조가 다르므로 설비 비율이 다름. 아래 그래프처럼 설비 비율이 다른 상태에서 각각 설비 상태 비율(가동률)을 측정하였다. 설비 상태 비율을 보면 시나리오 2가 ACTIVE상태 비율이 가장 높고 PAUSE상태 비율이 가장 낮음을 알 수 있다. 즉 시나리오 2가 가동률이 높다는 걸 알 수 있다. 또한 시나리오 4는 시나리오 1에서 STOCK가 1개와 STOCK을 연결하는 TRACK 1개가 추가된 구조이다. 시나리오 4의 경우 시나리오 1에 비해 ACTIVE상태가 훨씬 높아졌으며 PAUSE상태가 훨씬 낮아진 것을 알 수 있다.

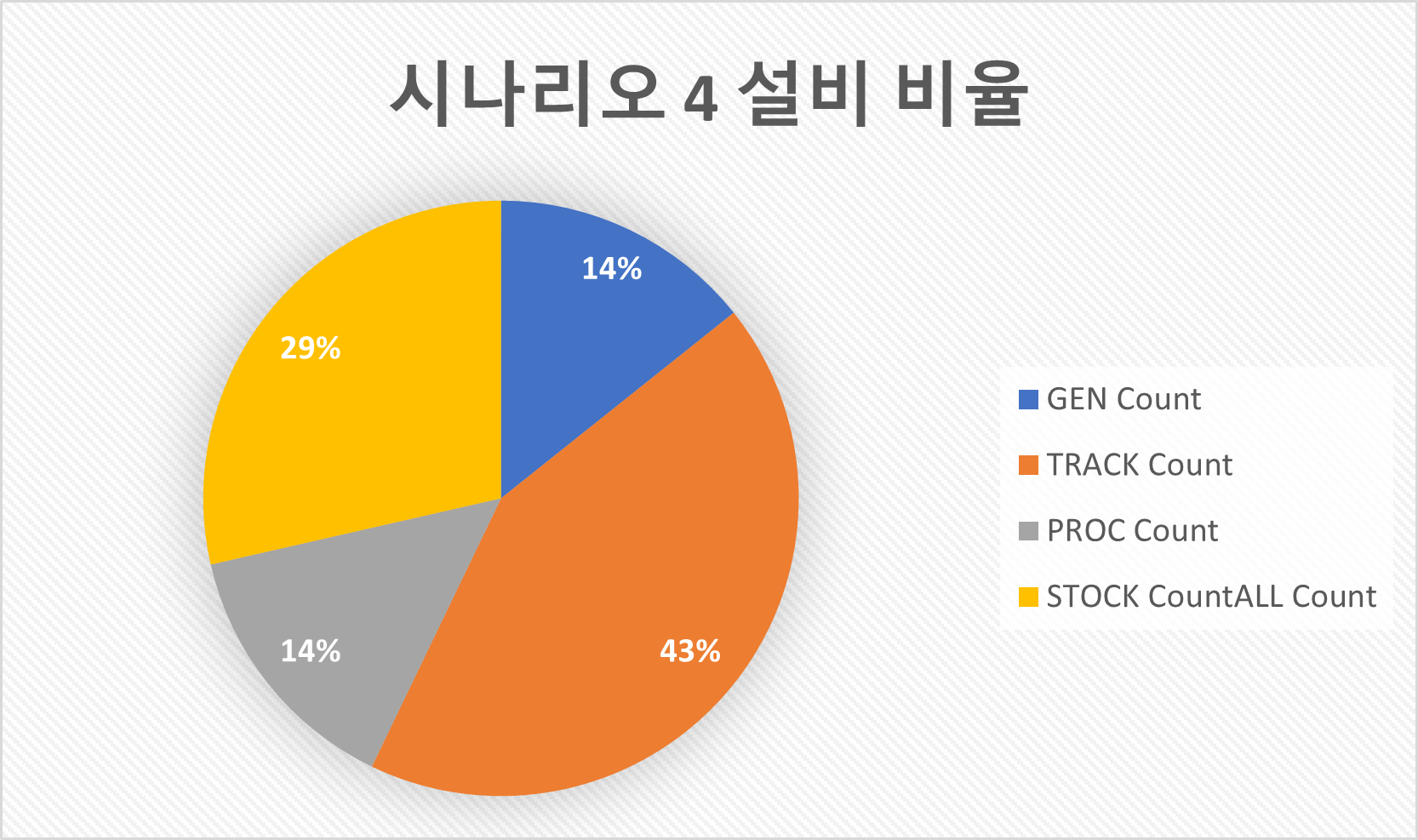
* 각 시나리오별 설비 비율

차트, 파이 차트이(가) 표시된 사진

자동 생성된 설명차트, 파이 차트이(가) 표시된 사진

자동 생성된 설명차트, 파이 차트이(가) 표시된 사진

자동 생성된 설명

****

* 설비 버퍼 평균

차트이(가) 표시된 사진

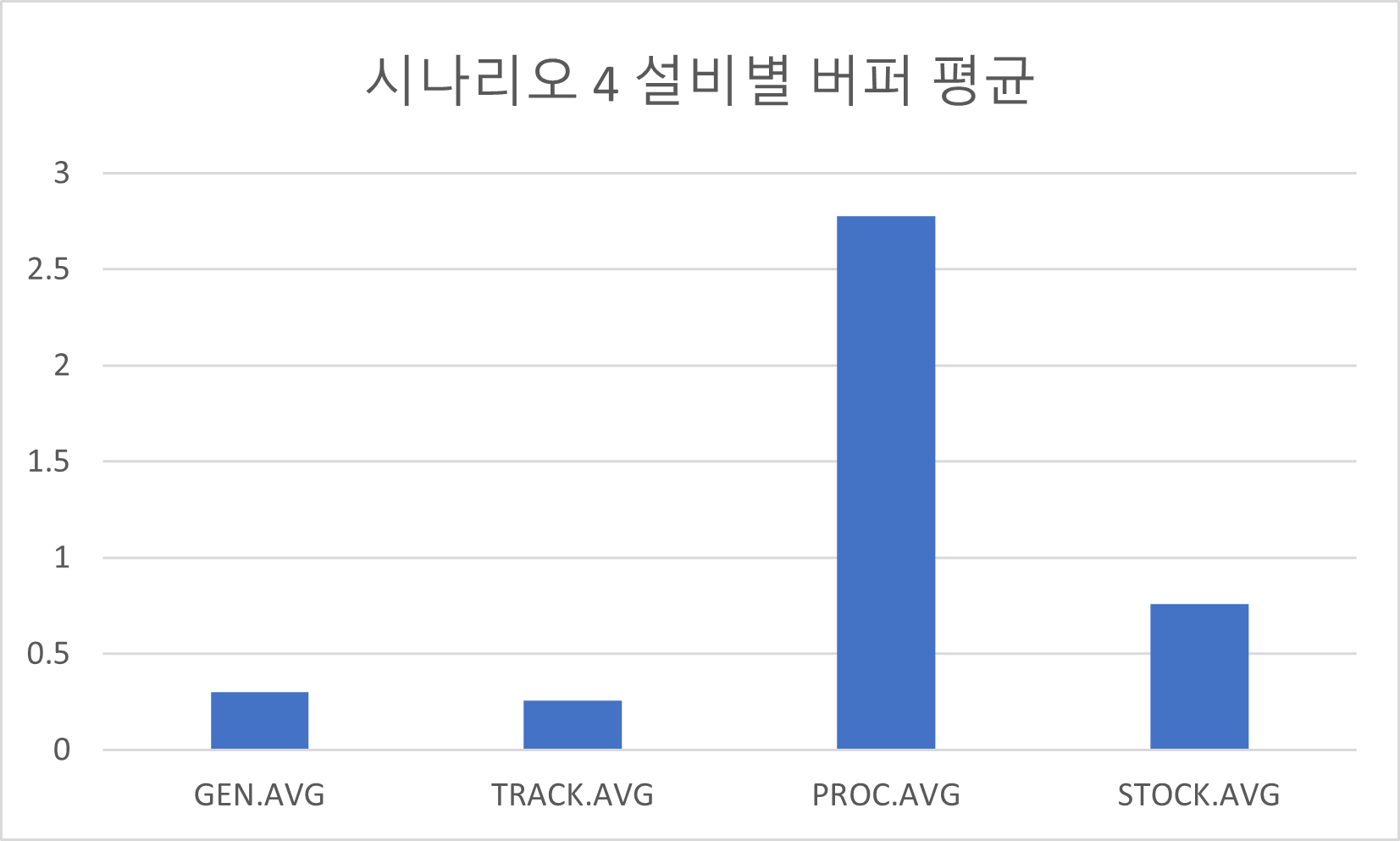
자동 생성된 설명

**차트이(가) 표시된 사진

자동 생성된 설명**

**차트이(가) 표시된 사진

자동 생성된 설명**

****

설비별 버퍼 평균에서는 전체적으로 봤을 때 시나리오 2번이 버퍼 평균이 적은걸 알 수 있음. 그러므로 시나리오 2가 효율적으로 동작함. 또한 시나리오 1과 시나리오 1에 STOCK과 그것을 연결하는 TRACK를 추가한 시나리오 4의 설비별 버퍼 평균을 비교하면, 4번 시나리오가 1번 시나리오보다 버퍼 평균이 적은 것을 알 수 있다.

1. **결론**

시나리오 2가 가동률, 생산량 모두 효율적으로 가동되었음을 알 수 있다. 그래서 STOCK이 많을수록 가동률 생산량의 변화를 확인하기 위해 1번에서 STOCK 개수를 변경한 4번 시나리오를 만들었고 4번 시나리오가 1번 시나리오에 비해 생산량이 많고 ACTIVE 가동률도 높아서 1번 시나리오보다 효율적인 공정이라는 것을 알 수 있었다. 이로인해, STOCK 개수가 많을수록 가동률/생산량이 효율적으로 가동한다는 결론을 내린다.