

Trabajo Fin de Grado  
Grado en Ingeniería de Organización Industrial

Estudio Comparativo de Sistemas de Control de  
Producción basados en Tarjetas: Kanban y  
COBACABANA

Autor: Mario Gabriel Edreira Ramírez

Tutor: Antonio Plácido Moreno Beltrán

Dep. Organización Industrial y Gestión de Empresas I  
Escuela Técnica Superior de Ingeniería  
Universidad de Sevilla

Sevilla, 2018





Trabajo Fin de Grado  
Grado en Ingeniería de Organización Industrial

# Estudio Comparativo de Sistemas de Control de Producción basados en Tarjetas: Kanban y COBACABANA

Autor:

Mario Gabriel Edreira Ramírez

Tutor:

Antonio Placido Moreno Beltrán

Profesor ayudante doctor

Dep Organización Industrial y Gestión de Empresas I

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2018



Proyecto Fin de Carrera: Estudio Comparativo de Sistemas de Control de Producción basados en Tarjetas:  
Kanban y COBACABANA

Autor: Mario Gabriel Edreira Ramírez

Tutor: Antonio Plácido Moreno Beltrán

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2018

El Secretario del Tribunal



# Agradecimientos

---

*A mi tutor Antonio Placido por, instruirme, guiarme y ayudarme, constantemente para la elaboración de este trabajo.*

*A mis familiares y pareja por apoyarme y darme todas las facilidades para poder desarrollar este proyecto.*



# Resumen

---

En la actualidad, en la literatura de los sistemas de producción tipo pull, como son Kanban y Cobacabana, se encuentra bastante información sobre ellos, en cuanto a cómo funcionan, sus características particulares, como se llegó a ellos, incluso comparaciones cualitativas entre ellos. Sin embargo, lo que se quiere en este estudio, es realizar un contraste cuantitativo.

Para ello, se va a simular, mediante el software Arena, cada uno de los sistemas. Se explicará cómo funcionan los módulos de este software que han sido usados para la realización de los modelos. Así como, se desarrollará paso a paso como se han modelado cada uno de ellos. Con ello se hará una recopilación de datos, de los que serán interesantes tres indicadores, que son el tiempo de ciclo promedio total, el tiempo de, el WIP promedio y la utilización de las estaciones de trabajo.

Con este estudio, se demuestra que el sistema que mejor rendimiento obtiene en cada uno de los escenarios es Cobacabana, a pesar de sus contras. En muchos casos, los resultados son muy parecidos a Kanban.



# Abstract

---

Currently, in the literature of pull-type production systems, such as Kanban and Cobacabana, there is enough information about them, in terms of how they work, their particular characteristics, how they were arrived at, even qualitative comparisons between them. However, what is wanted in this study is to make a quantitative contrast.

To do this, each one of the systems will be simulated using the Arena software. It will explain how the modules of this software that have been used for the realization of the models work. As well as, it will be developed step by step as each of them has been modeled. This will make a collection of data, which will be interesting three indicators, which are the total average cycle time, the time, the average WIP and the use of work stations.

With this study, it is shown that the system that performs best in each of the scenarios is Cobacabana, despite its cons. In many cases, the results are very similar to Kanban.

# Índice

---

<b>Agradecimientos</b>	<b>7</b>
<b>Resumen</b>	<b>9</b>
<b>Abstract</b>	<b>11</b>
<b>Índice</b>	<b>12</b>
<b>Índice de Tablas</b>	<b>14</b>
<b>Índice de Figuras</b>	<b>15</b>
<b>Capítulo 1 Objeto y Justificación</b>	<b>19</b>
<b>Capítulo 2 Trasfondo histórico de sistemas de control de la producción para entornos de producción tipo taller</b>	<b>22</b>
<b>Capítulo 3 Descripción del sistema de control Kanban</b>	<b>29</b>
<b>Capítulo 4 Descripción del sistema Cobacabana</b>	<b>33</b>
<b>Capítulo 5 Simulación en Software Arena</b>	<b>38</b>
<b>Capítulo 6 Modelado de los sistemas Kanban</b>	<b>55</b>
<b>Capítulo 7 Modelado de los sistemas Cobacabana</b>	<b>59</b>
<b>Capítulo 8 Discusión de Resultados</b>	<b>64</b>
<b>Capítulo 9 Conclusiones y líneas futuras</b>	<b>83</b>
<b>Bibliografía</b>	<b>84</b>

# TABLAS SEGÚN CAPÍTULOS

---

## **Capítulo 6**

Tabla 6-1. Tipos producto

## **Capítulo 7**

Tabla 7-1. Tipos producto

## **Capítulo 8**

Tabla 8-1. Tipos escenarios

Tabla 8-2. Calculos WIP Kanban escenario tipo 1

Tabla 8-3. Tabla resumen resultados Kanban escenario tipo 1

Tabla 8-4. Calculos WIP Cobacabana escenario tipo 1

Tabla 8-5. Tabla resumen resultados Cobacabana escenario tipo 1

Tabla 8-7. Calculos WIP Kanban escenario tipo 2

Tabla 8-8. Tabla resumen resultados Kanban escenario tipo 2

Tabla 8-9. Calculos WIP Cobacabana escenario tipo 2

Tabla 8-10. Tabla resumen resultados Cobacabana escenario tipo 2

Tabla 8-11. Calculos WIP Kanban escenario tipo 3

Tabla 8-12. Tabla resumen resultados Kanban escenario tipo 3

Tabla 8-13. Calculos WIP Cobacabana escenario tipo 3

Tabla 8-14. Tabla resumen resultados Cobacabana escenario tipo 3

Tabla 8-15. Calculos WIP Kanban escenario tipo 4

Tabla 8-16. Tabla resumen resultados Kanban escenario tipo 4

Tabla 8-17. Calculos WIP Cobacabana escenario tipo 4

Tabla 8-18. Tabla resumen resultados Cobacabanaescenario tipo 4

Tabla 8-19. Tabla resumen resultados

Tabla 8-20. Tabla Leyendas

## FIGURAS SEGÚN CAPÍTULOS

---

### Capítulo 2

Ilustración 2-1 Sistema Kanban de dos tarjetas. Hopp y Spearman (2004)

### Capítulo 3

Ilustración 3-2 Sistema Kanban de dos tarjetas. Hopp y Spearman (2004)

Ilustración 3-3 Equivalencia del sistema Kanban de una y dos tarjetas. Hopp y Spearman (2004).

### Capítulo 4

Ilustración 4-4 Bucle de tarjetas de aceptación COBACABANA (M. J. Land, 2009)

Ilustración 4-5 Display de tarjetas de aceptación COBACABANA (M. J. Land, 2009)

Ilustración 4-6 Captación y liberación de tarjetas en COBACABANA (M. J. Land, 2009)

Ilustración 4-7 Display de tarjetas libres, indicador de la situación de taller actual (M. J. Land, 2009)

### Capítulo 5

Ilustración 5-8 Interface Rockwell

Ilustración 5-9 Menú opciones del Módulo Create

Ilustración 5-10 Menú opciones del Módulo Dispose

Ilustración 5-11 Menú opciones del Módulo Process

Ilustración 5-12 Menú opciones del Módulo Decide

Ilustración 5-13 Menú opciones del Módulo Assign

Ilustración 5-14 Menú opciones del Módulo Entity  
Ilustración 5-15 Menú opciones del Módulo Queue  
Ilustración 5-16 Menú opciones del Módulo Resource  
Ilustración 5-17 Menú opciones del Módulo Set  
Ilustración 5-18 Menú opciones del Módulo Variable  
Ilustración 5-19 Menú opciones del Módulo Signal  
Ilustración 5-20 Menú opciones del Módulo Hold  
Ilustración 5-21 Menú opciones del Módulo Route  
Ilustración 5-15 Menú opciones del Modulo Station

## **Capítulo 6**

Ilustración 6-22. Modelo Completo de Kanban  
Ilustración 6-23. Modulo de entrada  
Ilustración 6-24. Módulo de procesado  
Ilustración 6-25. Linea de producción en estación 1  
Ilustración 6-26. Módulo de Salida  
Ilustración 6-27. Tablero de control

## **Capitulo 7**

Ilustración 7-28 Modelo Completo de Cobacabana  
Ilustración 7-29. Modulo de entrada  
Ilustración 7-30. Módulo de procesado  
Ilustración 7-31. Linea de producción en estación 1  
Ilustración 7-32. Módulo de Salida  
Ilustración 7-6 Modulo demonio, simulador de revisor  
Ilustración 7-7 Tablero de control de cargas

## **Capitulo 8**

Ilustración 8-33 Tiempo de ciclo Kanban Escenario tipo 1  
Ilustración 8-34 Number Busy Kanban escenario tipo 1  
Ilustración 8-35 Number in Queue Kanban escenario tipo 1  
Ilustración 8-36 Utilización Kanban escenario tipo 1  
Ilustración 8-5 Tiempo de ciclo Cobacabana Escenario tipo 1  
Ilustración 8-6 Number Busy Cobacabana escenario tipo 1  
Ilustración 8-7 Number in Queue Cobacabana escenario tipo 1  
Ilustración 8-8 Utilización Cobacabana escenario tipo 1  
Ilustración 8-9 Tiempo de ciclo Kanban Escenario tipo 2

Ilustración 8-10 Number Busy Kanban escenario tipo 2

Ilustración 8-11 Number in Queue Kanban escenario tipo 2

Ilustración 8-12 Utilización Kanban escenario tipo 2

Ilustración 8-373 Tiempo de ciclo Cobacabana Escenario tipo 2

Ilustración 8-14 Number Busy Cobacabana escenario tipo 2

Ilustración 8-15 Number in Queue Cobacabana escenario tipo 2

Ilustración 8-16 Utilización Cobacabana escenario tipo 2

Ilustración 8-387 Tiempo de ciclo Kanban Escenario tipo 3

Ilustración 8-18 Number Busy Kanban escenario tipo 3

Ilustración 8-19 Number in Queue Kanban escenario tipo 3

Ilustración 8-20 Utilización Kanban escenario tipo 3

Ilustración 8-21 Tiempo de ciclo Cobacabana Escenario tipo 3

Ilustración 8-392 Number Busy Cobacabana escenario tipo 3

Ilustración 8-240 Number in Queue Cobacabana escenario tipo 3

Ilustración 8-241 Utilización Cobacabana escenario tipo 3

Ilustración 8-25 Tiempo de ciclo Kanban Escenario tipo 4

Ilustración 8-426 Number Busy Kanban escenario tipo 4

Ilustración 8-27 Number in Queue Kanban escenario tipo 4

Ilustración 8-28 Utilización Kanban escenario tipo 4

Ilustración 8-29 Tiempo de ciclo Cobacabana Escenario tipo 4

Ilustración 8-30 Number Busy Cobacabana escenario tipo 4

Ilustración 8-431 Number in Queue Cobacabana escenario tipo 4

Ilustración 8-32 Utilización Cobacabana escenario tipo 4

# Capítulo 1: Objeto y Justificación.

## Introducción.

Actualmente la sociedad se encuentra inmersa en un entorno de demanda muy cambiante en la que los sistemas productivos tienen que adaptarse rápidamente a dichos cambios. Por tanto, son necesarias adoptar medidas que incrementen su eficiencia y productividad. Esta problemática tiene especial interés en entornos de tipo taller (job-shop), caracterizados por ser especialmente compleja su gestión.

Uno de los aspectos donde todavía existe un gran margen de mejora es en el ámbito del control de la producción en este tipo de entornos, ya que habitualmente son pocos los métodos desarrollados para ser aplicados a entornos de tipo taller. Este hecho está indicado en el trabajo de (Stevenson et al. 2005).

Hay que tener en cuenta que en los entornos de producción de tipo serie (flow-shop), son numerosas las referencias a diferentes implementaciones de sistemas de control de la producción, en su mayor parte a través de sistemas pull, controlados por tarjetas o Kanbans (ver para más detalle (González R. et al., 2011)). Sin embargo la implementación de los sistemas de control de la producción en entornos tipo taller es realizada habitualmente mediante algún sistema tipo push, donde se controla la carga de las diferentes estaciones (ver por ejemplo el estudio de implementación del sistema Work Load Control, WLC (L. Hendry et al., 2008)). La implementación de otro tipo de mecanismos (como los sistemas basados en tarjetas) no ha tenido, a día de hoy, éxito en este tipo de entornos. Ver por ejemplo el trabajo (Mark Stevenson et al., 2011) donde se discute la poca aplicabilidad de sistemas basados en tarjetas como los sistemas CONWIP (Constant Work In Process), POLCA (Paired cell Overlapping Loops of Cards with Authorization) o Kanban en entornos de tipo taller. Ninguno de los sistemas basados en tarjetas actualmente disponibles cumple los requisitos dinámicos necesarios para el control de la producción en entornos de tipo taller.

Recientemente, (M. J. Land, 2009), desarrolla un nuevo sistema de control basado en tarjetas denominado COBACABANA (Control of Balance by Card-Based Navigation), sistema que trata de incorporar la dinámica del sistema WLC mediante el uso de tarjetas o kanbans. Tras la publicación del citado artículo no se ha encontrado nueva documentación o publicación que determine la idoneidad del sistema COBACABANA para el control de la producción en entornos de tipo taller. Este es el principal motivo por el que se desarrolla el presente trabajo.

## Objetivos

El objetivo general de este proyecto es la simulación mediante el software Arena Rockwell de dos sistemas de control de producción, los cuales son: Kanban y Cobacabana. Mediante esta simulación, se recopilarán datos de interés que permitan comparar los dos sistemas.

## Desglose de Objetivos por Capítulos

Objetivo Capítulo 2:

Trasfondo histórico de sistemas de control de la producción para entornos de producción tipo taller.

Objetivo Capítulo 3:

Descripción del sistema de control Kanban.

Objetivo Capítulo 4:

Descripción del sistema Cobacabana.

Objetivo Capítulo 5:

Herramienta Software Arena Rockwell.

Objetivo Capítulo 6:

Modelado de los sistemas Kanban.

Objetivo Capítulo 7:

Modelado de los sistemas Cobacabana

Objetivo Capítulo 8:

Análisis de resultados.

Objetivo Capítulo 9:

Conclusiones y líneas futuras.

### **Estructura del Documento.**

El presente documento consta de 9 capítulos en los que se trata el control de los entornos de producción tipo taller mediante tarjetas.

En el capítulo 2 se realiza una introducción histórica del sistema Push y Pull. Dentro de esta sección, se verán los orígenes de estos sistemas, las diferencias entre estos, y algunos sistemas que utilizan sus principios.

En el capítulo 3 se describe el sistema de control Kanban, (Matthias Thüerer, 2016). A lo largo del capítulo se definen los principales procesos y parámetros de operación. Finalmente se realiza un análisis del sistema de control.

En el capítulo 4 se describe el sistema de control Cobacabana, propuesto en el artículo (M. J. Land, 2009). El autor, trata de trasladar los principios del WLC a un sistema basado en tarjetas. A lo largo del capítulo se definen los principales procesos y parámetros de operación. Finalmente se realiza un análisis del sistema de control.

Para el análisis experimental de los sistemas de control, y su comparación con los resultados obtenidos por M. Land se utiliza el Software Rockwell Arena, cuyos principales componentes se exponen en el capítulo 5.

Las soluciones y propuestas de modelado de los sistemas de control Kanban son desarrolladas en el capítulo 6.

En el capítulo 7 se encuentran las soluciones y propuestas de modelado de los sistemas de control Cobacabana.

En el capítulo 8 se puede ver la discusión de resultados. Aquí se encuentran los diferentes

## **Objeto y Justificación.**

---

resultados obtenidos para cada uno de los escenarios y para cada tipo de sistema de control, Kanban y Cobacabana. Se muestra además una explicación de los factores y parámetros utilizados para realizar estas simulaciones y encontramos una tabla resumen en la que se puede observar cómodamente todos los resultados para realizar la comparación de los sistemas.

En el capítulo 9 se exponen las conclusiones derivadas del presente trabajo y se proponen nuevas líneas de investigación para trabajos futuros.

En un último apartado se recoge la bibliografía consultada necesaria para la elaboración del presente trabajo.

## Capítulo 2: Trasfondo histórico de sistemas de control de la producción para entornos de producción tipo taller

### 2. Una breve historia de Pull

#### 2.1 Primero se encuentra MRP

Para entender sistemas PULL antes vamos a hablar sobre lo que se conoce como su antepasado, MRP. Previo al dominio del ordenador en la fabricación, el inventario era controlado usando métodos tipo ROP/ROQ (reorder—point/reorder-quantity). En los 60, Joseph Oricly, Oliver Wigth y George Plossl, junto con otros investigadores, desarrollaron un nuevo sistema, al que denominaron, Material Requirements Planning (MRP). Después de un comienzo lento, en los 70, MRP comenzó a tener mayor uso debido al AMERICAN PRODUCTION AND INVENTORY CONTROL SOCIETY. En 1971 Orickly detectó unas aproximadamente 150 implantaciones del sistema, pero en 1981, la cifra ascendía a 8000. En los 80, MRP evolucionó a MANUFACTURING RESOURCES PLANNING (MRP II), que combinaba MRP con Master Scheduling, Rough-cut, Capacity Planing, entre otros módulos.

Mientras que MRP iba dominando la producción americana, la historia iba tomando un curso diferente en Japón. Varias compañías japonesas, destacando Toyota, desarrollaron los métodos antiguos (ROP/ROQ) a alto nivel. Empezando en los 40, Taiicho Ohno comenzó un sistema que permitiría a Toyota competir con los fabricantes americanos, pero no lo iba a conseguir debido a que Toyota no tenía medios para soportar los volúmenes de producción con los que estaban trabajando las entidades americanas. Este enfoque, es ahora conocido como Toyota Production System diseñado para hacer las cosas lo mejor posible y en un flujo continuo.

Según Ohno, el sistema de producción de Toyota descansa sobre dos pilares: autonomation y producción Just-in-Time. Autonomation, automatización con toques humanos, es determinar la forma óptima de realizar una tarea determinada y luego hacer el estándar de “mejores prácticas”. Autonomation involucra “fool proofing” que usa dispositivos para verificar rápidamente dimensiones y otros atributos de calidad para permitir a los trabajadores ser sus propios responsables de calidad. Si se detectan problemas, la línea de producción se detiene hasta que los problemas se corrigen. Esto elimina la necesidad de rehacer las líneas y también eliminaba mucho desperdicio.

El sistema de producción de Toyota fue denominado 5S: SEIRI, SEITON, SEISO, SEIKETSU y SHITSUKE que eran técnicas organizacionales y de gestión interna destinadas al logro de la autonomation y el control visual.

De acuerdo con Ohno, JIT, involucra dos componentes, Kanban y nivel de producción. Kanban o “Pull production” se convirtió en el sello característico del sistema de producción de Toyota al punto de que muchas veces eran ‘sinónimos’.

La revolución del JIT fue realmente una evolución, los resultados fueron sustanciales. En 1945, las configuraciones tomaban de dos a tres horas, en cambio, en 1962, conllevaba apenas 15 minutos y en 1971 algunos setups habían bajado a 3 minutos.

A principio de los 80, las fabricas americanas, eran conscientes de que se habían quedado atrás en la innovación a nivel de fabricación e incluso en la eficiencia especialmente en el sector automovilístico. Sin embargo, las ventas de MRP seguían subiendo, pero muchos, empezaron a pensar que implantar MRP había sido un error. En 1980, encuestas mostraban que menos del 10% de las firmas, habían recuperado sus inversiones en al menos dos años. Es entonces cuando JIT empieza a ser bien visto.

Directores americanos fueron rápidamente cautivados con todo lo Japonés. Profesores Americanos fueron a Japón a aprender de primera mano, lo que estaba sucediendo y por supuesto, redactaron libros. El primero en ser publicado fue en 1981, Driving the Productivity Machine: Production and Control in Japan. Seguido por Japanese Manufacturing Techniques: Nine lessons in Simplicity entre otros. A pesar del revuelo por el interés en JIT en todo, la industria americana de los 80 era una mezcla. Las empresas que implementan JIT se enfrentan con una filosofía engañosamente simple y un complicado conjunto de técnicas. Los gerentes tuvieron que contribuir con una considerable innovación site-specific para producir sistemas viables. Dependiendo de cuan perspicaces fuesen usarían JIT o no.

Para finales de los 80, JIT comenzó a ser eclipsado por un gran sistema, Enterprise Resources Planning (ERP). Con el desarrollo de la arquitectura tecnológica cliente/servidor, se hace factible la integración virtual de todo el negocio de la corporación en una base de datos. ERP ofreció integración near-total y software best of breed en aplicaciones específicas. Cabe mencionar que ERP era más complejo que MRPII, conteniendo módulos para todas las funciones de negocio imaginables y comprensivamente más caras.

Como ERP empezó a crecer, parecía que JIT mantenía su camino. Aunque Toyota seguía en la cima en la industria automovilística debido a su calidad y métricas de eficiencia, el interés en su sistema, estaba disminuyendo. Tras la publicación de The machine that changed the world, donde se comparaban técnicas europeas, americanas y japonesas en la industria del automóvil, llegaba a la conclusión de que los métodos japoneses eran ampliamente superiores. Además, los autores renovaron el concepto de JIT por lo que se conoce actualmente como Lean Manufacturing. Con un nuevo nombre, el interés por el sistema creado por Taiichi Ohno comenzó a ser un tema fuertemente mencionado en el mundo de la industria. JIT dio lugar también a un movimiento que finalmente se haría más grande que este, conocido como Total Quality Management (TQM). Este movimiento fue impulso a Motorola a establecer una ambiciosa meta de calidad y a desarrollar un conjunto de técnicas estadísticas para la medirla y lograrla. Este enfoque es lo que se conoce como Six Sigma.

A pesar de tomar distintos caminos, parece que Lean Manufacturing y los movimientos Six Sigma están a punto de fusionarse, como sugieren el reciente best-sellers en administración, Lean Six Sigma. Al mismo tiempo, el enfoque computacional característico de MRP/ERP se está consolidando. Después del Y2K (problema del 2000), el termino ERP fue rápidamente reemplazado por SCM (Supply Chain Management). SAP, el mayor proveedor del mundo de

software ERP, cambio su línea entera de productos en cuestión de meses. Debido a que ambas, ERP y SCM trazan sus raíces en la producción cuantitativa y el control de inventario, quizás no sea sorprendente que puedan tener los mismos productos de software. Sin embargo, todo esto conlleva a preguntarnos si algún día, tendrá lugar un movimiento Lean Six Sigma SCM.

## 2.2 Pull

Las corporaciones americanas trataban de navegar en el océano de los nuevos conceptos y comprender el sistema de producción de Toyota junto con el concepto de Pull. Como con cualquier nueva tendencia en el ámbito de la dirección, muchos investigadores empezaron a trabajar en JIT. Ahora bien, en 1977, aparece un documento describiendo Kanban, publicado por investigadores japoneses, con el título de 'Toyota Production System and Kanban System: Materialization of Just-in Time and Respect-For-Human System'. Pasados unos años después, otros investigadores japoneses, Kimura y Terada, publican otros documentos en los que se describían las mecánicas de Kanban y los requisitos para su implementación. Para principios de los 80, ya teníamos muy bien descritas las mecánicas de Kanban. Esto representaba el primer sistema denominado bajo el término "Pull".

### 2.2.1 Kanban explicación inicial

La versión clásica de Kanban, de Toyota, es llamado sistema "two-card kanban" y se muestra en la siguiente figura:

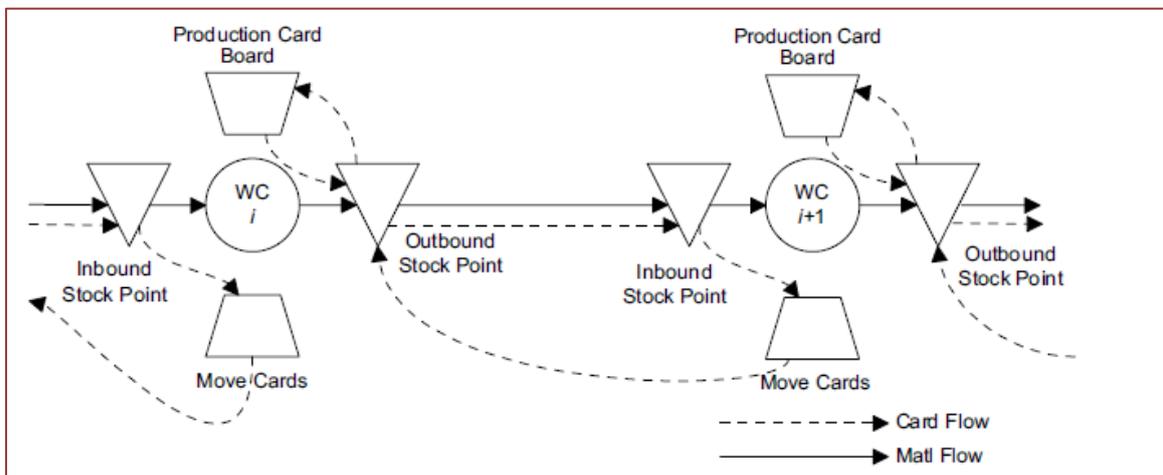


Ilustración 2-1 Sistema Kanban de dos tarjetas. Hopp y Spearman (2004)

La producción comienza cuando el controlador de materiales con una tarjeta de movimiento coge un contenedor estándar de stock de salida. La tarjeta de movimiento autoriza a ese controlador a tomar estas partes y le dice dónde se necesitan las piezas. Antes de que se coja el contenedor, la tarjeta de producción se quita del contenedor estándar y se coloca en el tablero de tarjetas de producción. La producción no puede comenzar sin una tarjeta de producción, un contenedor apropiado de partes entrantes y una estación de trabajo inactiva. Cuando los tres están disponibles, el trabajador quita un contenedor estándar de stock de entrada, elimina la tarjeta de movimiento del contenedor, la coloca en la tolva para mover

tarjetas en la estación de trabajo, y comienza a procesar las partes. Periódicamente un manejador de materiales recoge la (s) tarjeta (s) de movimiento, localiza las partes necesarias, las transporta a la estación de trabajo, y el proceso se repite en la próxima estación de trabajo aguas arriba.

### 2.2.2 ¿Por qué tanto revuelo con sistemas Pull?

Si Kanban no es más que un sistema de producción en serie con bloqueo, entonces ¿Qué lo hace especial?

Los sistemas con bloqueo se producen naturalmente ¿Por qué Toyota restringe deliberadamente el flujo en su sistema? Los investigadores y profesionales han tenido que abordar esta pregunta fundamental desde los inicios de JIT.

El primer paso, y el más fácil, para entender Kanban en específico, y Pull en general, es caracterizar sus beneficios entre los que encontramos:

- Reducido WIP y tiempo de ciclo.
- Coste reducido.
- Flujo de producción más suave.
- Calidad mejorada.

También hay que identificar de qué se trata el Kanban que produce estos beneficios. Esto es importante si queremos entender cuanto podemos simplificar o generalizar Kanban y que siga siendo eficaz. En esencia, esta es la pregunta que determina lo que realmente constituye el Pull.

La analogía del agua y las rocas sugiere que los beneficios de Kanban provienen de mejoras ambientales (eliminando "rocas" o problemas). Alguna investigación ha apoyado esta vista. Por ejemplo, Huang (1983) realizó una simulación de Kanban y concluyó que sin cambios significativos en el entorno de fabricación, Kanban no sería exitosa en los Estados Unidos. Krajewski (1987) realizó una simulación aún más completa y concluyó que el beneficio de Kanban se debe más a un entorno mejorado que a cualquier otro cambio en la logística.

Sin embargo, atribuir los beneficios de Kanban a mejoras ambientales no explica exactamente las mejoras que produce. Además, la mayoría de los investigadores no estaban dispuestos a acreditar mejoras del entorno como única explicación de todos los beneficios de JIT. Ohno consideraba Kanban parte esencial para JIT, y muchas empresas tenían versiones que reportaron gran éxito. Por otra parte, si el método de control de flujo no tenía mucha importancia, luego lo tuvo. Las compañías que aplicaban JIT podrían haber revertido a MRP después de mejorar sus entornos. Sin embargo, esto no ocurrió. Por lo tanto, para entender mejor lo que hace Kanban, los investigadores recurrieron modelos matemáticos.

Karmarkar (1986, 1991) parece haber sido el primero en observar las similitudes de Kanban con el stock de base en dos documentos simples. El sistema que discute es muy similar al sistema

tradicional básico de stock descrito por Simpson (1958). Describe el sistema de stock básico de la siguiente manera:

Cuando se realiza un pedido, se llena del inventario si este no es cero. Si el inventario es cero, la orden se coloca en un archivo pendiente, para ser archivada cuando un artículo llega. En cualquier caso, una orden de fabricación se coloca inmediatamente para producir un artículo con fin de reemplazar el que ha sido consumido. El operario, a su vez, inmediatamente realiza un pedido de la materia prima requerida y tan pronto se completa este pedido, procede a producir la orden requerida. De esta forma, una orden contra el último inventario de un artículo terminado se transmite inmediatamente a lo largo de la línea a todas las operaciones de fabricación.

Sin embargo, aunque Kanban y los sistemas de stock de base son similares, no son idénticos. La diferencia fue descrita por Spearman (1992) de la siguiente manera: Kanban no haría un pedido de más partes si se demanda cuando no hay stock en el punto de salida de este. En cambio, habría una o más tarjetas de producción ya en proceso y cada vez que uno de estos fuese completado, la tarjeta de movimiento en espera sería inmediatamente adjuntada al contenedor de partes y la tarjeta de producción serían enviados de vuelta a producción. De esta manera, un sistema Kanban limita la cantidad de WIP que puede haber en el sistema, mientras que el sistema de stock básico no.

Spearman (1990) y junto con Zazanis (1992) encontraron que si bien las mejoras en entorno son ciertamente influyentes, hay tres principales razones logísticas para la mejora de rendimiento de sistemas de Pull:

1. Menos congestión: la comparación de una red de cola abierta con una "equivalente" cerrada, muestra que el WIP promedio es menor en la red cerrada que en la red abierta dado el mismo rendimiento. El efecto es relativamente menor y se debe al hecho de que las longitudes de las colas no tienen correlación en un sistema abierto pero están correlacionados negativamente en una red de colas cerrada.

2. Control más fácil: este es un beneficio fundamental que resulta de varias observaciones:

- WIP es más fácil de controlar que el rendimiento porque se puede observar directamente.

- El rendimiento generalmente se controla con respecto a la capacidad. Porque no se puede observar directamente, la capacidad debe estimarse considerando tiempo de proceso, tiempo de configuración, interrupciones aleatorias, eficiencia del trabajador, re trabajo y otros factores que afectan el tasa de producción potencial.

- El rendimiento se controla especificando una tasa de entrada. Si la tasa de entrada es menor que la capacidad de la línea, el rendimiento es igual a la entrada. Si no, el rendimiento es igual a la capacidad y las construcciones de WIP. Al estimar la capacidad de manera incorrecta, se puede exceder fácilmente la verdadera capacidad. Esto es particularmente cierto cuando se buscan altas tasas de utilización. Como resultado, los sistemas que controlan WIP

son sustancialmente más robustos para controlar los errores que los sistemas que controlan el rendimiento.

3. WIP Cap: los beneficios de un entorno Pull son en gran parte por el hecho de que el WIP está limitado. Se observa que un límite global simple en el WIP promoverá los mismos beneficios que los citados para Kanban y mostrando que el rendimiento de una red de colas cerrada sin bloqueo es mayor que el de una red de colas cerrada con bloqueo.

### 2.3 Entendiendo Push y Pull

Para iniciar, asumiremos que Kanban es un sistema tipo Pull, mientras que MRP es un sistema tipo Push. Para definir Pull, necesitamos caracterizar cual es la diferencia entre estos dos sistemas.

Axsäter y Rosling, muestran que MRP es más general que una política de stock. Esto implica que MRP podría incluso dominar la política Kanban porque ésta es simplemente una política de stock con un número límite de pedidos (según estos autores). Sin embargo, la restricción adicional parece ser la clave de un sistema Pull. El hecho de que Kanban no continúe agregando órdenes al sistema, pone un límite natural en el WIP. Como la salida de una línea de producción es una función creciente, limitada por el WIP, si el tiempo de flujo comienza a crecer, cuando llegue a cierto punto, no tiene sentido agregar más trabajo a un sistema que ya está saturado.

Un sistema de producción Pull es aquel que limita explícitamente la cantidad de trabajo en proceso que puede estar en el sistema. Por defecto, esto implica que un sistema de producción Push es aquel sin límite explícito en la cantidad de trabajo en proceso que puede estar en el sistema.

En el mundo real nunca encontraremos sistemas Pull o Push puros. Por ejemplo, mientras que el sistema Kanban establece un claro límite en el WIP mediante el uso de tarjetas de producción, existen muchas circunstancias bajo las cuales este límite, será anulado. Por otro lado, si MRP no establece un límite en el WIP, hay un nivel que hará a la administración ignorar los lanzamientos planificados o revisar el plan maestro de producción para así evitar un mayor nivel de WIP. Resumiendo, hay un nivel de WIP, una especie de límite, para todos los sistemas. La distinción, sin embargo, es que el límite de WIP que se usa en los sistemas pull es generalmente pequeño. Cualquier límite de WIP que se realice en un sistema Push está implícito, es grande y usualmente entra en juego demasiado tarde.

Pull puede ser implementado de varias maneras. Kanban es una manera para limitar el WIP, pero existen otras. Vamos a ver los sistemas más comunes en la industria y en la literatura y lo clasificaremos como Pull o Push.

- MRP. Sistema push. El número de lanzamientos está establecido según el plan maestro de producción sin tener en cuenta el estado del sistema. A priori no hay un límite en el WIP.
- Sistema Clásico Kanban. Sistema pull. El número de tarjetas Kanban establece un límite en el WIP.

- MRP con restricción de WIP. Sistema Pull.
- CONWIP es un sistema Pull porque limita el WIP a través de tarjetas similares a Kanban.
- Los sistemas POLCA propuestos por Suri (1998) son sistemas Pull porque, como Kanban y CONWIP, WIP es limitado por tarjetas.

## Capítulo 3: Descripción del sistema de control Kanban.

### Sistema Kanban

Taiichi Ohno introdujo los sistemas Kanban en Toyota, fue el primer sistema de control de producción basado en tarjetas denominado sistema Pull. Estos sistemas fueron originalmente desarrollados para coordinar la confluencia de diferentes flujos de productos, ya sea en el interior de una empresa como entre diferentes empresas (Thürer et al, 2016).

Para llevar a cabo esta coordinación, hace uso de un dispositivo de señalización que, normalmente, son tarjetas (la traducción de Kanban en japonés es tarjeta o boleto), que no tienen por qué ir unidas a un solo producto, y pueden adjuntarse también a contenedores en los que caben varios productos. A cada lote se le asigna una tarjeta Kanban en cada punto en el tiempo, por lo que el número de tarjetas Kanbans limita el stock de materiales.

La tarjeta Kanban enumera las características clave del material al cual está anexada. Casi siempre estos datos incluyen (Chapman, 2006, Lödding, 2013):

- Número e identificación del componente.
- Ubicación dentro del almacén.
- Tamaño del contenedor (en caso de que el material esté almacenado en un contenedor).
- Centro de trabajo (o proveedor) de origen.

Se utilizan principalmente dos tipos de tarjetas, pero de cada tipo puede haber varias en un mismo proceso, estas son (Velasco y Campins, 2013):

1. Tarjeta de transporte: especifica el tipo y cantidad de producto a retirar del proceso anterior.
2. Tarjeta de producción: indica el tipo y la cantidad a fabricar.

Hay instalaciones en las que usan el sistema Kanban, pero en vez de hacer uso de tarjetas emplean otros tipos de señalización. Por ejemplo, un hueco en el suelo del taller es indicio suficiente de que hace falta el siguiente contenedor o una señal como una bandera para avisar de que es el momento para el próximo contenedor (Heizer y Render, 2008).

Ohno (1988) estableció un conjunto de reglas Kanban, que son las siguientes (Thürer et al, 2016):

1. La última línea acude a la anterior para recoger los productos que necesita.
2. La línea anterior produce ciertos artículos en la cantidad indicada por la tarjeta Kanban.
3. No se hacen artículos o se transportan sin un Kanban.

4. Siempre hay que adjuntar un Kanban al producto.

La versión clásica de Kanban como pionera de Toyota es el llamado “Sistema Kanban de dos tarjetas”. En este sistema se hace uso de los dos tipos de tarjetas explicados anteriormente. La producción comienza cuando un trabajador que tiene una tarjeta de movimiento se dispone a coger un contenedor del punto de salida. Esta tarjeta autoriza al trabajador a tomar este contenedor y le dice dónde se va a necesitar. Antes de su retirada, la tarjeta de producción que lleva adjunta el contenedor debe ser liberada. Entonces el trabajador llevara el contenedor al punto de entrada de la siguiente estación de trabajo, en la que no podrá ser procesado el producto hasta que no haya una tarjeta de producción libre. En el momento que se tenga, se liberará la tarjeta de movimiento y se le adjuntará la de producción. Periódicamente, un trabajador recoge las tarjetas de movimiento, localiza las piezas necesarias, las trasporta a la estación de trabajo y el proceso se repite en la estación de trabajo posterior. Para clarificar este concepto ver figura 3-1:

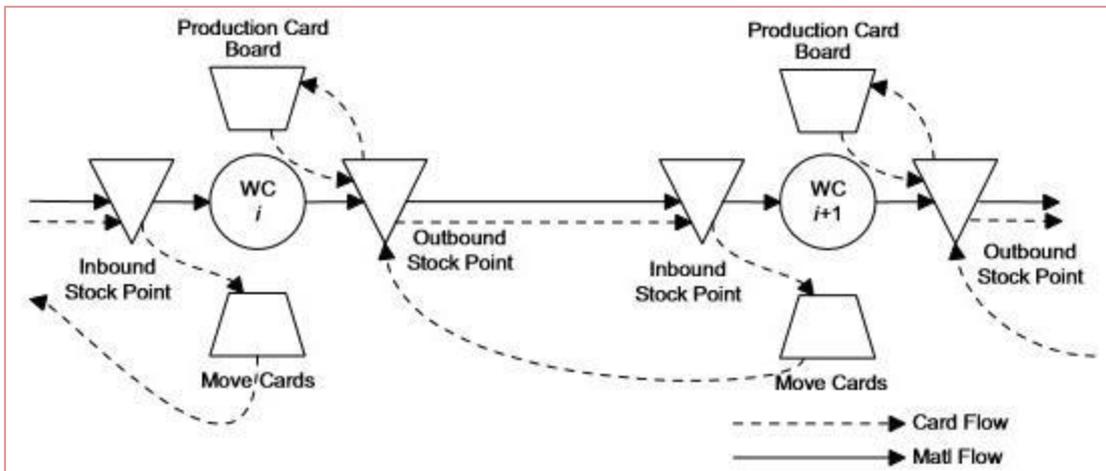


Ilustración 3-1 Sistema Kanban de dos tarjetas. Hopp y Spearman (2004)

Como podemos observar, este sistema marca una diferencia entre el procesamiento de las piezas y el movimiento del material. Si queremos convertir el sistema Kanban de dos tarjetas, en un sistema de una tarjeta, basta con tratar el movimiento del contenedor como un proceso, por lo que se utilizaría una tarjeta de producción. Lo podemos ver en la siguiente figura, en la que también se muestra que un sistema Kanban con una tarjeta de producción es equivalente a un sistema de producción en serie con bloqueo, ver figura 3-2, (Hopp y Spearman, 2004).

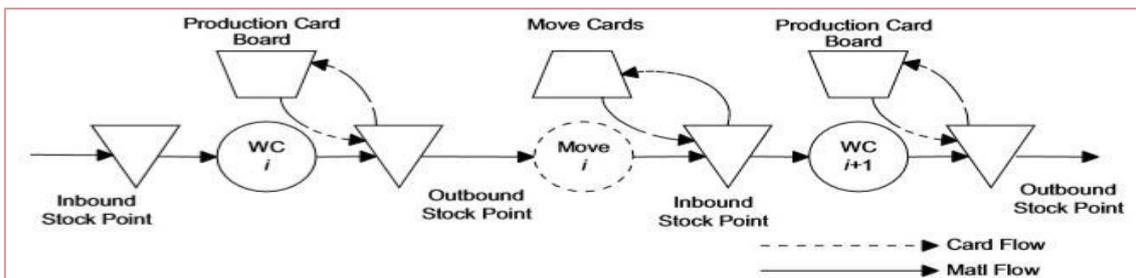


Ilustración 3-2 Equivalencia del sistema Kanban de una y dos tarjetas. Hopp y Spearman (2004).

En este caso, el contenedor para empezar a ser procesado necesita una tarjeta de producción. Una vez se le ha asignado la tarjeta, se realiza el proceso que indica y pasará a la cola del siguiente, siempre con la tarjeta adjunta, hasta que el siguiente proceso tenga una tarjeta libre. Entonces devolverá la tarjeta del proceso anterior y se le adjuntará la de este.

Al establecer un sistema de control Kanban es necesario determinar la cantidad de tarjetas Kanban requeridas. Para calcularlas se parte de las siguientes magnitudes:

- Q: Consumo medio previsto del material por unidad de tiempo (por ejemplo, número de piezas por día).
- LT: Tiempo de entrega total de suministro del material que figura en cada tarjeta, desde que se solicita y se envía vacío al proceso anterior, hasta que se recibe lleno, pasando por el periodo de producción o suministro dentro del proceso anterior. La unidad de medida del tiempo LT será la utilizada para Q.
- q: Cantidad de material asociado a cada tarjeta, reunido en un recipiente (por ejemplo, un contenedor).
- K: Cantidad total de tarjetas Kanban que podrán existir entre las de tipo transporte y las de producción. La distribución entre ellas dependerá del proceso (si hay o no producción en el proceso al que se solicita material), el número de componentes y productos procesados y las cantidades que figuren en las tarjetas.

Es evidente que debe cumplirse que:

$$K=Q \times LT q$$

Si se considera la existencia de un stock de seguridad SE, este stock dará lugar a una cantidad adicional de tarjetas:

$$K_e=SEq$$

En caso de que se exprese como un porcentaje sobre el número de tarjetas calculadas anteriormente, a través de un coeficiente  $\mu$  será:

$$K_e=\mu * Q * LTq$$

El total de tarjetas necesarias será pues:

$$K_T=K+ K_e = Q *LT *(1+ )$$

Evidentemente, la cantidad de producto en curso dependerá de la cantidad de tarjetas que existan en los casilleros, de forma que cuando no haya ninguna tarjeta, la cantidad de producto en curso será la máxima posible.

Por otra parte, se observa que para reducir el producto en curso WIP debe actuarse sobre el lead time LT o sobre el stock de seguridad SE (o lo que es lo mismo, sobre  $\mu$ ), de forma que cuanto más se reduzcan, menor podrá ser dicho stock. Como esto es altamente deseable, convendrá tener en cuenta cómo reducir los tiempos y la seguridad al mínimo.

Esta fórmula es útil, pero en la práctica se suele preferir comenzar el proceso con suficiente número de tarjetas, y sentirse cómodos, de forma que se pueda hacer frente a cualquier incertidumbre. Es una buena forma de empezar a implantar el método siempre que no permanezca esa comodidad mucho tiempo. Se necesita ir reduciendo el inventario para que se expongan los problemas y se solucionen lo más pronto posible (Chapman, 2006, Cuatrecasas, 2011).

Hay varias cosas a tener en cuenta respecto a los sistemas Kanban, que son (Thürer et al, 2016, Heizer y Render, 2008):

- Cada paso de ruta tiene que ser representado por un bucle Kanban. Esto significa que la variabilidad de ruta debe ser baja para evitar un gran número de bucles superpuestos que de otro modo serían engorrosos para controlar.

Modelado y estudio comparativo de sistemas de control de producción tipo pull.

- Los sistemas Kanban en cuanto al problema de control de inventarios, son muy efectivos si las estaciones están desacopladas. Sin embargo, para el control de pedidos surgen varios problemas. Por ejemplo, la última estación que se utiliza para controlar el proceso requiere el mayor número de Kanbans, y los Kanbans permanecen en cada bucle, dificultando la propagación de la información específica del pedido.
- Los contenedores suelen ser muy pequeños, normalmente representan el trabajo de unas pocas horas de producción.
- Un sistema como éste requiere una programación muy ajustada.
- Hay que producir pequeñas cantidades varias veces al día.
- El proceso debe funcionar con suavidad, con poca variabilidad en el plazo de producción o aprovisionamiento, porque cualquier falta de suministros tiene una repercusión casi inmediata en todo el sistema.
- El Kanban pone mayor énfasis en cumplir la programación, en reducir el tiempo y el coste de las preparaciones o cambios en las máquinas y en una manipulación económica de los materiales.

## Capítulo 4: Descripción del sistema Cobacabana

### Sistema Cobacabana.

El sistema COBACABANA se basa en los principios del WLC, haciendo uso de tarjetas para tomar decisiones respecto a las órdenes de aceptación y las órdenes de liberación. Habitualmente los sistemas Kanban limitan la cantidad de WIP asociando a cada tarjeta un trabajo o conjunto de trabajos (contenedores). De esta manera se limita el WIP máximo en el sistema, ya que el número máximo de trabajos en el sistema tiene que ser inferior al número de tarjetas. Sin embargo la aplicación en entornos con múltiples tipos de producto, donde cada producto tiene asociado un tiempo de proceso, hace que la utilización de tarjetas en este sentido no sea del todo eficiente, ya que se asigna la misma importancia a los diferentes trabajos independientemente de su tiempo de proceso. Esto puede dificultar el control sobre el propio sistema, ya que falsea la carga real del sistema, al no existir una equivalencia real de ocupación de las máquinas con el número de tarjetas. Este hecho se agrava además cuando existen diferentes rutas para cada tipo de producto. En este sentido, el sistema COBACABANA está diseñado bajo la perspectiva de asociar a cada tarjeta un determinado valor de carga (tiempo en que la máquina estará ocupada con el trabajo). De esta manera para cada centro de trabajo que ha de procesar un trabajo, éste tomará del panel de control las tarjetas equivalentes a la carga de trabajo. En el caso de no haber tarjetas disponibles para procesar el trabajo en algún centro, dicho trabajo tendrá que esperar a que se liberen las tarjetas suficientes para que el trabajo pueda ser procesado. Obviamente el número de tarjetas disponibles en el panel de control está relacionado con la norma que limita la carga en dicho centro.

El número de tarjetas del centro de trabajo  $s$  que debe asociarse al trabajo  $j$  queda determinado por la siguiente ecuación. La justificación y proceso detallado de obtención de la misma puede analizarse en el artículo (M. J. Land, 2009):

$$A_{js} = \frac{p_{js}}{O_s^{*D} / T_s^{*D}}$$

Se establece que el factor  $d_{jst} * p_{js}$  determina la contribución de carga del trabajo  $j$  en el centro de trabajo  $s$  antes de ser liberado, obsérvese la no equivalencia entre el factor de depreciación  $d_{jst}$  y el factor  $\frac{1}{O_{st}^{*D} / T_{st}^{*D}}$ . Tal y como puede analizarse en el artículo (Martin Land, 2004) el factor  $d_{jst}$  es mayor cuanto más cercana esté la estación de trabajo a la primera operación de la ruta. Sin embargo el factor  $\frac{1}{O_{st}^{*D} / T_{st}^{*D}}$ , no depende de la posición que ocupe la estación de trabajo  $s$  en la ruta.

Respecto a los sistemas de liberación:

- El valor de  $\frac{1}{O_{st}^{*D}/T_{st}^{*D}} * p_{js}$  determinaría el número de tarjetas correspondientes al centro de trabajo  $s$  que se asignarían al trabajo  $j$ . Las tarjetas corresponden a un valor de tiempo determinado.
- El valor de  $A_s^H$  o  $A_s^A$  (según se use el método A o el B) determinará el número total de tarjetas que tendrá el display del centro de trabajo  $s$ .

Los requerimientos que se le hacen al sistema COBACABANA son los siguientes:

- debe servir como soporte para la organización de la producción en planta
- debe servir de soporte para la toma de decisiones respecto a la aceptación de nuevos trabajos
- debe indicar fechas de entrega realistas en el momento de aceptación del trabajo

### Órdenes de aceptación y compromiso de fecha de entrega.

Una de las ventajas de este tipo de sistemas, es que, a pesar de operar en entornos job-shop, permite estimar el tiempo de finalización de los trabajos, con lo que de cara a los clientes se puede aproximar una fecha de entrega. El tiempo mínimo de entrega puede determinarse a través del tiempo de permanencia estimado en el pool,  $T_j^{*P}$  desde la aceptación hasta la liberación y la suma de los tiempos de proceso estimados en la ruta  $T_s^{*D}$  ):

$$d_j^{min} = T_j^{*P} + \sum_{s \in S_j} T_s^{*D}$$

La estación con mayor cantidad de trabajos en espera en el pool es la que determinará el tiempo en espera previo a la liberación.

De cara a la aceptación de órdenes de trabajo se establece un indicador numérico  $A_{js}$  que determina la cantidad de tarjetas que deben retirarse en caso de aceptar un trabajo.

En este caso, las tarjetas utilizadas para la toma de decisiones entorno a las de órdenes de aceptación representarán una fracción de la jornada laboral.

En la ilustración 4-1 se muestra el bucle de operación para la aceptación de trabajos. Una vez se acepta el trabajo, se envían las tarjetas correspondientes al pool. Una vez liberado el trabajo a planta, las tarjetas de aceptación correspondientes se devuelven al display de tarjetas de aceptación.

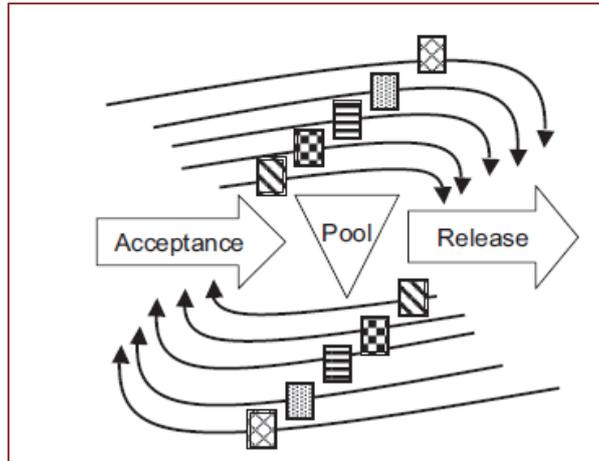


Ilustración 4-1 Bucle de tarjetas de aceptación COBACABANA (M. J. Land, 2009)

En la ilustración 4-2 se observa un ejemplo de display de tarjetas de aceptación, las tarjetas retiradas nos muestran el tiempo que falta para que se pueda liberar un trabajo que incorpore en su ruta a la estación de trabajo que se esté analizando. Por lo tanto, para un trabajo concreto, el tiempo de liberación ( $T_j^{*P}$ ) se determinará en función al mayor de los tiempos de liberación de entre las distintas estaciones que componen su ruta de proceso. Es decir, si estuviéramos analizando la liberación de un trabajo cuya ruta estuviera compuesta por los centros de trabajo 2 y 3, el tiempo de espera hasta su liberación estaría determinado por el display del centro de trabajo 3 al ser este el que tiene menos tarjetas, o lo que es lo mismo, mayor carga de trabajo entre todos los trabajos que hay en ese instante en la zona de espera.

A partir de este display, podremos determinar entonces la propuesta de tiempo de entrega y determinar la aceptación o no del trabajo. El tiempo de entrega se determina según la ecuación 6; en función de la ruta del trabajo, se determinará  $T_j^{*P}$  teniendo en cuenta el display del centro de trabajo de la ruta con el mayor número de tarjetas retiradas, tal y como se ha explicado anteriormente.

El trabajo será aceptado, y por lo tanto enviado a la zona de espera, en el caso de que esta estimación del tiempo de entrega sea inferior al tiempo de entrega máximo solicitado por el cliente.

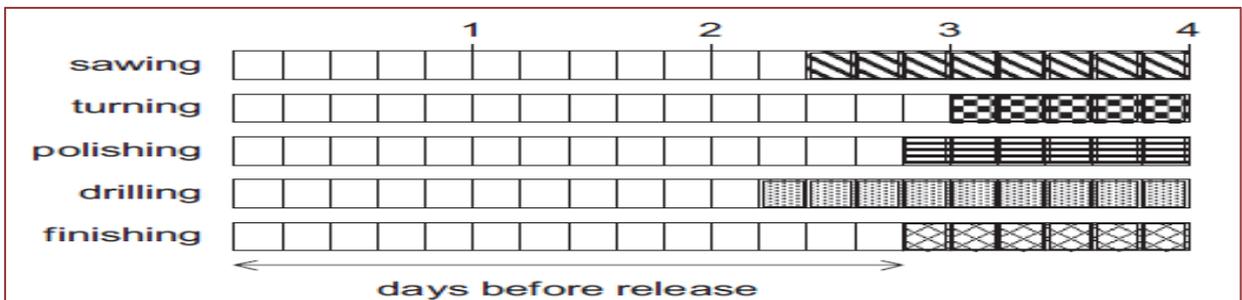


Ilustración 4-2 Display de tarjetas de aceptación COBACABANA (M. J. Land, 2009)

## Órdenes de liberación y control de planta.

El funcionamiento del sistema de órdenes de liberación y control de planta es el siguiente. Una vez que el trabajo es liberado (ilustración 4-3) se envían las tarjetas correspondientes a su carga de trabajo a las estaciones de trabajo que componen su ruta, al mismo tiempo, una vez que el trabajo concluya en una estación, la tarjeta correspondiente debe abandonar la estación de trabajo y volver al display del planificador.

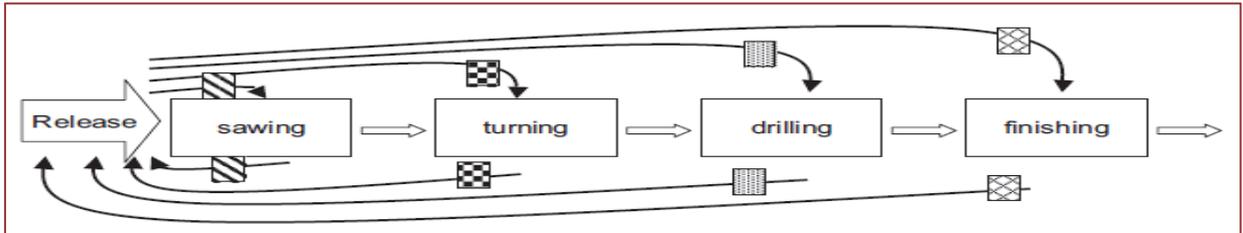


Ilustración 4-3 Captación y liberación de tarjetas en COBACABANA (M. J. Land, 2009)

En la ilustración 4-4 se muestra un ejemplo acerca de un panel de control o display que emplea como referencia divisiones del 5% de la carga máxima (norma) que admite cada centro de trabajo. La forma de dividir el panel no es detallada por el autor del sistema, ni se discute la posible repercusión que tendrá en los resultados.



Ilustración 4-4 Display de tarjetas libres, indicador de la situación de taller actual (M. J. Land, 2009)

El sistema de liberación es sencillo, el trabajo del pool será liberado si una vez determinadas sus contribuciones a las cargas de cada estación observamos que existen tarjetas disponibles que contengan la suficiente carga de trabajo para todas y cada una de las estaciones de la ruta de trabajo. Una vez que los trabajos son liberados en planta, estos se pueden ordenar mediante FCFS (first-come-first-serve) o de acuerdo a las fechas de entrega.

## Análisis previo del sistema de control COBACABANA.

Como se ha comentado anteriormente el sistema COBACABANA trata de emular el sistema WLC a través de un sistema controlado por tarjetas. Para ello, para cada estación de trabajo en la ruta de un pedido se traducen las aportaciones de carga del trabajo en número de tarjetas a asignar al mismo. De este modo los criterios de liberación de un trabajo al taller se basan en la existencia o no de tarjetas libres para un trabajo concreto.

Como se ha visto en capítulos anteriores, los sistemas WLC tratan de mantener una longitud de colas estable de cara a amortiguar la influencia de las variaciones en la función de entrada de pedidos, así como impedir la falta de alimentación de las estaciones o centros de trabajo.

Esta característica es muy importante en los sistemas WLC, ya que los métodos de liberación basados en la carga de trabajo ponderan la aportación de carga del trabajo  $j$  a la estación  $s$  en función a la cercanía o lejanía temporal de dicho trabajo respecto al tiempo estimado de llegada a la estación. Es decir, se considera la probabilidad de que un trabajo pase durante el transcurso dos procesos de liberación (entre liberaciones) de una estación a la siguiente. Por ello la estimación de carga de las estaciones es muy sofisticada, en el sentido de que se estiman futuras cargas y no sólo la carga directa que presenta el sistema en un instante determinado.

Sin embargo, en la propuesta del sistema COBACABANA resulta cuestionable la atemporalidad de las tarjetas asignadas, es decir, no existe diferenciación entre el número de tarjetas del centro de trabajo  $s$  a asignar al trabajo  $j$  en función al periodo estimado en que el trabajo  $j$  será procesado en la estación  $s$ . Dicho de otro modo, el número de tarjetas de la estación  $s$  será el mismo si dicha estación de trabajo es la primera de la ruta o la tercera.

Bajo estas circunstancias, podría darse el caso extremo de asignar todas las tarjetas de la estación de trabajo  $s$  a pedidos cuyo paso por dicha estación se produciría 5 o 6 periodos de liberación posteriores al de liberación, e impedir el uso de dicha estación durante todo ese tiempo.

Este hecho hace pensar que el comportamiento del sistema COBACABANA, a priori, sea pobre comparado con los sistemas clásicos WLC.

## Capítulo 5: Simulación en Software Arena.

El análisis del sistema de control COBACABANA conlleva la necesidad de poner en marcha un diseño experimental basado en la simulación de la producción tipo job-shop al objeto de obtener una serie de resultados tratados estadísticamente con los que poder realizar un estudio comparativo respecto a los resultados obtenidos por M. Land en su tesis (Martin Land, 2004).

El software seleccionado para llevar a cabo la simulación de los distintos sistemas de control a analizar es Arena Simulation, los criterios utilizados para esta selección son:

- Fácil identificación de los módulos proceso del software con los elementos del proceso a simular.
- Posibilidad y facilidad de análisis de sensibilidad de los sistemas de control.
- Tratamiento estadístico de los resultados por parte del software.
- Conocimientos previos del programa por parte del alumno.

### Introducción al software Arena Simulation

Arena es un potente software de modelado y simulación de diferentes áreas de negocio. Se ha diseñado para analizar el impacto de los cambios que suponen los complejos y significativos rediseños asociados a la cadena de suministros, procesos, logística, distribución y almacenaje y sistemas de servicio. Tiene gran flexibilidad y cubre gran cantidad de aplicaciones a modelar con cualquier nivel de detalle o complejidad. (Bradley, 2007)

Un escenario típico incluye:

- Análisis detallado del tipo de sistema de manufactura, incluyendo el transporte manual de componentes.
- Análisis de servicio al cliente y sistemas de dirección orientados al cliente.
- Análisis de cadenas de suministro globales que incluyen almacenamiento, transporte y sistemas logísticos.
- Predicción del funcionamiento de sistemas en función de medidas clave como costes, tasa de salida de piezas, tiempos de ciclo y utilización.
- Identificación de los procesos cuello de botella como colas construidas con sobreutilización de recursos.
- Planificación del personal, equipos y requerimientos de material.”

Arena Software es un simulador intuitivo gracias a que la programación está basada en la colocación y unión gráfica de distintos módulos de proceso.

El entorno de Arena Simulation consta de tres zonas diferenciadas:

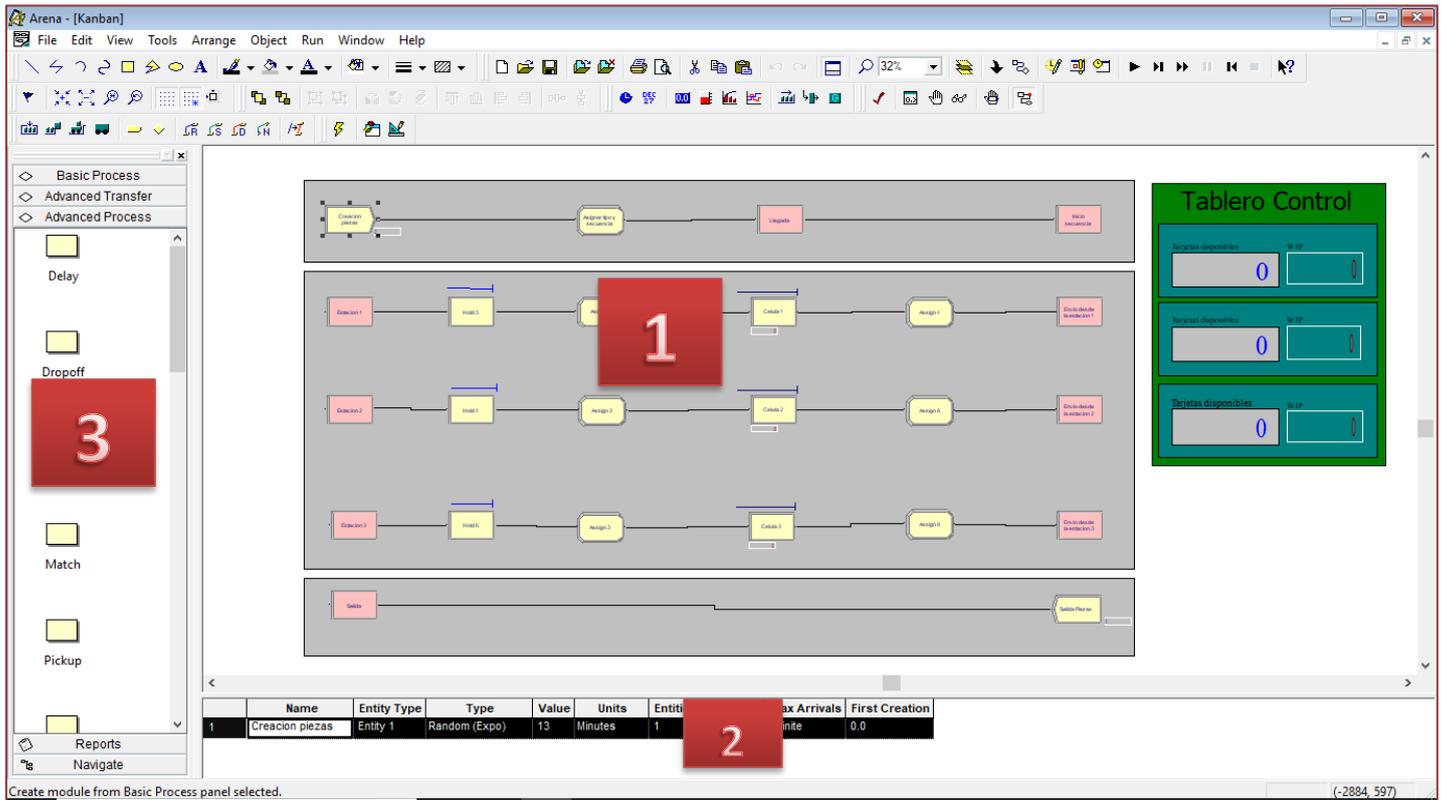


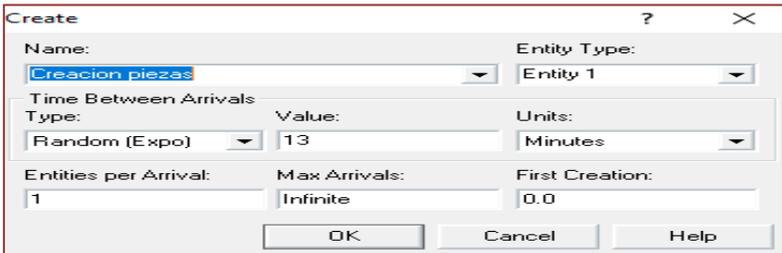
Ilustración 5-1 Interface Rockwell

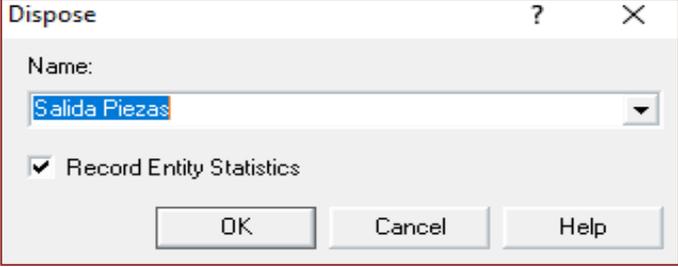
La construcción de un modelo es relativamente fácil pues una vez diseñado el diagrama de flujos y la secuenciación de los eventos discretos del proceso, basta con “arrastrar” módulos de la zona 3 a la zona 1 y darle valores a los parámetros correspondientes.

### Descripción de los módulos de proceso.

A continuación se realiza una descripción de los principales módulos de flujo y de datos del software según la guía de (Bradley, 2007).

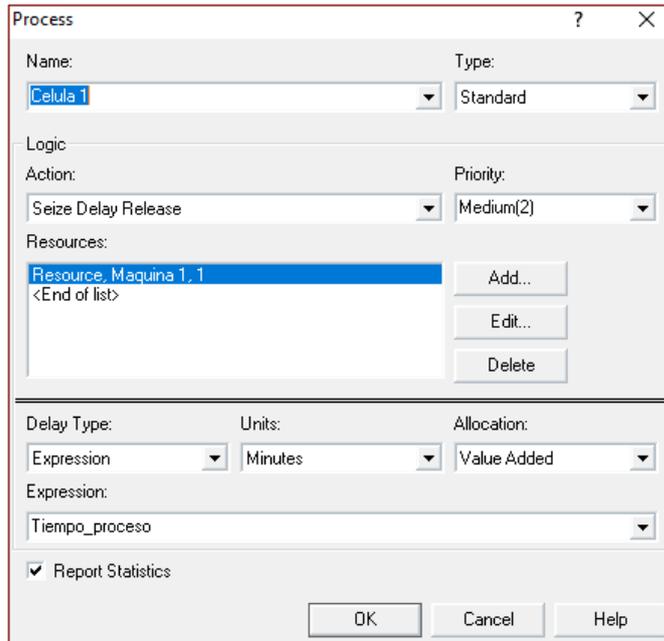
## Módulos de flujo. Basic Process

Módulo Create	
<p>Este módulo representa la llegada de entidades al modelo de simulación. Las entidades se crean usando una planificación o basándose en el tiempo entre llegadas. En este módulo se especifica también el tipo de entidad de que se trata. Una vez se incluye en el modelo a la derecha del símbolo aparece bajo una línea el número de entidades creadas.</p> <p><i>Posibles Usos.</i></p> <ul style="list-style-type: none"> <li>• Punto de inicio de producción en una línea de fabricación.</li> <li>• Llegada de un documento (por ejemplo, una petición, una factura, una orden) en un proceso de negocio.</li> <li>• Llegada de un cliente a un proceso de servicio (por ejemplo, un restaurante, una oficina de información).</li> </ul> <p><i>Parámetros.</i></p> <ul style="list-style-type: none"> <li>• Name: identificador único del módulo</li> <li>• Entity Type: nombre del tipo de entidad a ser generada.</li> <li>• Type: tipo de flujo de llegada a generar. Los tipos incluidos son: Random (usa una distribución exponencial y hay que indicar la media), Schedule (usa una distribución exponencial pero la media se determina a partir del módulo Schedule especificado), Constant (se especifica un valor constante), o Expresión (se puede elegir entre distintas distribuciones).</li> <li>• Value: determina la media de la distribución exponencial (si se usa Random) o el valor constante (si se usa Constant) para el tiempo entre llegadas.</li> <li>• Schedule Name: identifica el nombre de la planificación a usar. La planificación define el formato de llegada para las entidades que llegan al sistema. Sólo se aplica cuando se usa en Type, Schedule.</li> <li>• Expression: cualquier distribución o valor que especifique el tiempo entre llegadas. Se aplica sólo cuando en Type se usa Expression.</li> <li>• Units: unidades de tiempo que se usan para los tiempos entre llegadas y de la primera creación.</li> <li>• Entities per Arrival: número de entidades que se introducirán en el sistema en un momento dado con cada llegada.</li> <li>• Max Arrivals: número máximo de entidades que generará este módulo.</li> <li>• First Creation: momento de inicio en el que llega la primera entidad al sistema.</li> </ul>	
 <p style="text-align: center;"><b>Ilustración 5-2 Menú opciones del Módulo Create</b></p>	

<i>Módulo Dispose</i>	 Dispose
<p>Este módulo representa el punto final de entidades en un modelo de simulación. Las estadísticas de la entidad se registrarán antes de que la entidad se elimine del modelo.</p> <p><i>Posibles Usos.</i></p> <ul style="list-style-type: none"><li>• Partes que abandonan un servicio</li><li>• Finalización de un proceso de negocio</li><li>• Clientes abandonando un comercio</li></ul> <p><i>Parámetros.</i></p> <ul style="list-style-type: none"><li>• Name: identificador único del módulo.</li><li>• Record EntityStatistics: determina si las estadísticas de las entidades entrantes se registrarán o no. Estas estadísticas incluyen value-added time, nonvalue-added time, wait time, transfer time, other time, total time, valueadded cost, non-value-added cost, wait cost, transfer cost, other cost, y total cost.</li></ul>	
 <p><b>Ilustración 5-3 Menú opciones del Módulo Dispose</b></p>	

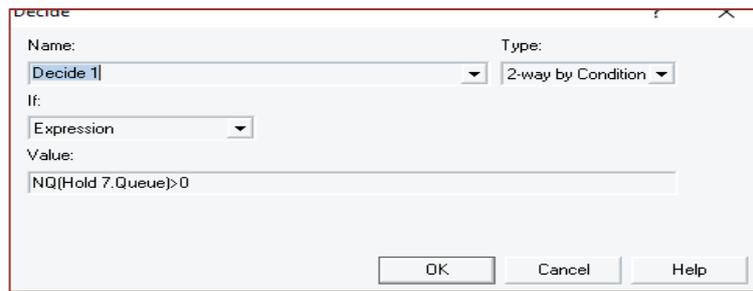
<i>Módulo Process</i>	 <b>Process</b>
<p>Este módulo corresponde a la principal forma de procesamiento en simulación. Se dispone de opciones para ocupar y liberar un recurso. Adicionalmente, existe la opción de especificar un “submodelo” y especificar jerárquicamente la lógica definida por el usuario. El tiempo de proceso se le añade a la entidad y se puede considerar como valor añadido, valor no-añadido, transferencia, espera u otros.</p> <p>Una vez se introduce en el modelo, aparece un número en la parte inferior del símbolo que indica el número de entidades que actualmente están procesándose.</p> <p><i>Posibles Usos.</i></p> <ul style="list-style-type: none"> <li>• Mecanizado de una parte.</li> <li>• Revisión de un documento para completarlo.</li> <li>• Rellenar órdenes.</li> <li>• Servir a un cliente.</li> </ul> <p><i>Parámetros</i></p> <ul style="list-style-type: none"> <li>• Name : identificador único del módulo</li> <li>• Type: método que especifica la lógica dentro del módulo. Un procesado Standard significa que toda la lógica se guardará dentro de un módulo Process y se definirá por una acción (Action) particular. Submodel indica que la lógica se definirá jerárquicamente en un “submodelo” que puede incluir un número indeterminado de módulos lógicos.</li> <li>• Action: tipo de proceso que tendrá lugar dentro del módulo. Existen cuatro tipos: Delay, Seize Delay y Seize Delay Release.</li> </ul> <p>Delay indica que solamente se llevará a cabo un proceso de retardo sin que existan restricciones de recursos. Seize Delay indica que un recurso será asignado en este módulo y que habrá un retardo y la liberación del recurso ocurrirá más tarde. Seize Delay Release indica que se asignará un recurso seguido por un retardo y luego, se liberará el recurso reservado. Delay Release indica que un recurso ha sido reservado previamente y que la entidad se retardará simplemente, y luego se liberará el recurso especificado.</p> <ul style="list-style-type: none"> <li>• Priority: valor de prioridad de la entidad que espera acceder en este módulo un determinado recurso si una o más entidades esperan el mismo recurso(s) en cualquier lugar en el modelo.</li> <li>• Resources: lista del recurso o conjunto de recursos utilizados para procesar la entidad. No se aplica cuando Action tiene el valor de Delay o cuando Type es submodel.</li> <li>• Delay Type: tipo de distribución o método de especificar los parámetros del retardo. Constant y Expression requieren valores simples, mientras que Normal, Uniform, y Triangular requieren varios parámetros.</li> <li>• Units: unidades de tiempo para los parámetros de retardo.</li> <li>• Allocation: determina cómo se asigna el tiempo de procesado y el coste del proceso a la entidad.</li> <li>• Minimum. valor mínimo en el caso de una distribución uniforme o triangular.</li> <li>• Value: valor medio para una distribución normal, el valor constante para un retardo de tiempo constante, o la moda para una distribución triangular.</li> </ul>	

- Maximum: valor máximo para una distribución uniforme o triangular.
- Std Dev: desviación estándar para una distribución normal.
- Expression: expresión cuyo valor se evalúa y se usa para el procesado del retardo de tiempo.

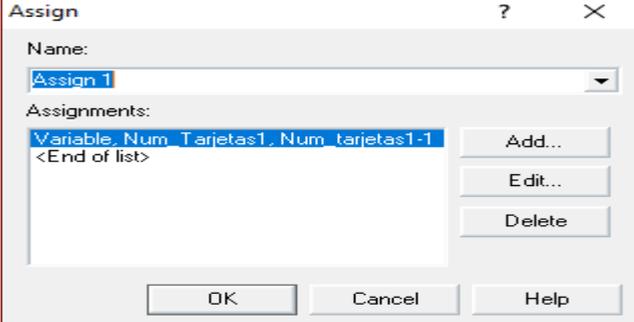


**Ilustración 5-4 Menú opciones del Módulo Process**

<i>Módulo Decide</i>	 Decide
<p>Este módulo permite a los procesos tomar decisiones en el sistema. Incluye la opción de tomar decisiones basándose en una o más condiciones (por ejemplo, si el tipo de la entidad es Gold Car) o basándose en una o más probabilidades (por ejemplo, 75% verdadero, 25% falso). Las condiciones se pueden basar en valores de atributos (por ejemplo, prioridad), valores de variables (por ejemplo, número de rechazados), el tipo de entidad o una expresión.</p> <p>Hay dos puntos de salida del módulo Decide cuando se especifica el tipo 2-way chance o 2-way condition. Hay un punto de salida para las entidades “verdaderas” y una para las entidades “falsas”. Cuando se especifica el tipo Nway chance o condition, aparecen múltiples puntos de salida para cada condición o probabilidad y una única salida “else”.</p> <p>Una vez incluido en el modelo, cerca de cada una de las ramas que salen del símbolo que representa el módulo, aparece un número. En la rama “True” corresponde al número de entidades que toman la rama de verdadero y el de la rama “False” el número de entidades que toman la rama de falso.</p> <p><i>Posibles Usos</i></p> <ul style="list-style-type: none"> <li>• Envío de partes defectuosas para que se vuelvan a hacer.</li> <li>• Ramas aceptadas frente a rechazadas.</li> <li>• Envío de clientes prioritarios a procesos dedicados</li> </ul> <p><i>Parámetros</i></p> <ul style="list-style-type: none"> <li>• Name: identificador único del módulo.</li> <li>• Type: indica si la decisión se basa en una condición o es aleatoria/porcentual. El tipo se puede especificar como 2-way o N-way. 2-way permite definir una condición o probabilidad (más la salida “false”). N-way permite definir cualquier número de condiciones o probabilidades, aparte de la salida “false”.</li> <li>• Conditions: define una o más condiciones que se usan para dirigir las entidades a los distintos módulos.</li> <li>• Percentages: define uno o más porcentajes usados para encaminar las entidades a los distintos módulos.</li> <li>• Percent True: valor que se comprobará para determinar el porcentaje de entidades que se han enviado a través de la salida True.</li> <li>• If: tipos de condiciones disponibles para ser evaluados.</li> <li>• Named: especifica el nombre de la variable, atributo, o tipo de entidad que se evaluarán cuando una entidad entre en el módulo.</li> <li>• Is: evaluador de la condición.</li> <li>• Value: expresión que se comparará con un atributo o variable o que se evaluará como una única expresión para determinar si es verdadero o falso.</li> </ul>	



**Ilustración 5-5 Menú opciones del Módulo Decide**

Módulo Assign	
<p>Este módulo se usa para asignar valores nuevos a las variables, a los atributos de las entidades, tipos de entidades, figuras de las entidades, u otras variables del sistema. Se pueden hacer múltiples asignaciones con un único módulo Assign. Para añadir una nueva variable al modelo, simplemente se selecciona Add, Type: Entity, el nombre de la variable y el valor que se desea tome a partir de ese momento.</p> <p>Posibles Usos</p> <ul style="list-style-type: none"> <li>• Acumular el número de subensamblados añadidos a una parte.</li> <li>• Cambiar el tipo de entidad para representar una copia de un formulario multicopia.</li> <li>• Establecer una prioridad del cliente.</li> </ul> <p>Parámetros</p> <ul style="list-style-type: none"> <li>• Name: identificador único del módulo.</li> <li>• Assignments: especifica la o las asignaciones que se llevarán a cabo cuando la entidad ejecute el módulo.</li> <li>• Type: tipo de asignación que se va a realizar. Other, puede incluir variables del sistema, tales como capacidad de los recursos o tiempo de finalización de la simulación.</li> <li>• Variable Name: nombre de la variable a la que se asignará un nuevo valor.</li> <li>• Attribute Name: nombre del atributo de la entidad al que se le asignará un nuevo valor.</li> <li>• Entity Type: nuevo tipo de entidad que se le asignará a la entidad cuando entre en el módulo.</li> <li>• Entity Picture: nueva imagen de la entidad que se le asignará.</li> <li>• Other: Identifica la variable del sistema especial a la que se le asignará un nuevo valor.</li> <li>• New value: Valor asignado al atributo, variable, u otras variables del sistema.</li> </ul>	
<div data-bbox="475 1256 1118 1592" style="border: 1px solid red; padding: 10px; text-align: center;">  <p><b>Ilustración 5-6 Menú opciones del Módulo Assign</b></p> </div>	

## Módulos de datos. Basic Process

<i>Módulo Entity</i>	 <b>Entity</b>																																																		
<p>Este módulo de datos define los diversos tipos de entidades y su valor de imagen inicial en la simulación.</p> <p><i>Posibles usos</i></p> <ul style="list-style-type: none"> <li>• Elementos que se van a producir o ensamblar (piezas, pallets).</li> <li>• Documentos: formularios, e-mails, faxes, informes...</li> <li>• Gente que se mueve a través del proceso (clientes).</li> </ul> <p><i>Parámetros</i></p> <ul style="list-style-type: none"> <li>• Entity Type: nombre de la entidad que se va a definir.</li> <li>• Initial Picture: representación gráfica de la entidad al inicio de la simulación.</li> <li>• Holding Cost/Hour: coste por horas de procesamiento de la entidad a lo largo del sistema. Este coste se sufre siempre que la entidad se encuentre en cualquier sitio del sistema.</li> <li>• Initial VA Cost: valor de coste inicial que se asignará al atributo value-added cost de la entidad. Este atributo acumula al coste sufrido cuando una entidad pasa tiempo en una actividad de espera; por ejemplo, esperando a ser metida en un lote o esperando un recurso(s) en un módulo Process.</li> <li>• Initial Transfer Cost: valor de coste inicial que se le asignará al atributo de coste de transferencia de la entidad. Este atributo acumula el coste sufrido cuando una entidad pasa tiempo en una actividad de transferencia.</li> <li>• Initial Other Cost: valor de coste inicial que se asignará al atributo other cost de la entidad. Este atributo acumula el coste sufrido cuando una entidad pasa tiempo en una actividad de transferencia.</li> </ul>																																																			
<table border="1" style="width: 100%; border-collapse: collapse; font-size: 8pt;"> <thead> <tr> <th></th> <th>Entity Type</th> <th>Initial Picture</th> <th>Holding Cost / Hour</th> <th>Initial VA Cost</th> <th>Initial NVA Cost</th> <th>Initial Waiting Cost</th> <th>Initial Tran Cost</th> <th>Initial Other Cost</th> <th>Report Statistics</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Bicicleta tipo 1</td> <td>Picture.Report</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td style="text-align: center;">✓</td> </tr> <tr> <td>2</td> <td>Bicicleta tipo 2</td> <td>Picture.Report</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td style="text-align: center;">✓</td> </tr> <tr> <td>3</td> <td>Bicicleta tipo 3</td> <td>Picture.Report</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td style="text-align: center;">✓</td> </tr> <tr> <td>4</td> <td>Pedidos</td> <td>Picture.Report</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td style="text-align: center;">✓</td> </tr> </tbody> </table> <p><b>Ilustración 5-7 Menú opciones del Módulo Entity</b></p>			Entity Type	Initial Picture	Holding Cost / Hour	Initial VA Cost	Initial NVA Cost	Initial Waiting Cost	Initial Tran Cost	Initial Other Cost	Report Statistics	1	Bicicleta tipo 1	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓	2	Bicicleta tipo 2	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓	3	Bicicleta tipo 3	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓	4	Pedidos	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓
	Entity Type	Initial Picture	Holding Cost / Hour	Initial VA Cost	Initial NVA Cost	Initial Waiting Cost	Initial Tran Cost	Initial Other Cost	Report Statistics																																										
1	Bicicleta tipo 1	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓																																										
2	Bicicleta tipo 2	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓																																										
3	Bicicleta tipo 3	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓																																										
4	Pedidos	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	✓																																										

<i>Módulo Queue</i>	 Queue																																										
<p>Este módulo de datos se puede usar para cambiar la regla para una determinada cola. La regla de la cola por defecto es First In, First Out salvo que se indique otra cosa en este módulo. Hay un campo adicional que permite definir la cola como compartida.</p> <p><i>Posibles usos</i></p> <ul style="list-style-type: none"> <li>• Cola de trabajos esperando un recurso en un módulo Process.</li> <li>• Área de almacenamiento de documentos que esperan ser cotejados en un módulo Batch.</li> </ul> <p><i>Parámetros</i></p> <ul style="list-style-type: none"> <li>• Name: nombre de la cola cuyas características se van a definir.</li> <li>• Type: regla de encolado para la cola, la cual puede estar basada en un atributo. Los tipos incluyen First In, First Out; Last In, First Out; Lowest Attribute Value (primero); y Highest Attribute Value (primero). Un valor de atributo bajo puede ser 0 o 1, mientras que un valor alto puede ser 200 o 300.</li> <li>• Attribute name: atributo que se evaluará para los tipos Lowest Attribute Value o Highest Attribute Value. Las entidades con valores de atributos más bajos o más altos serán encoladas primero en la cola.</li> <li>• Shared: campo de selección que determina si una determinada cola se usa en múltiples sitios dentro del modelo de simulación. Sólo se puede usar en el caso de recursos de petición (es decir, con el módulo Seize del panel Advanced Process).</li> </ul>																																											
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 25%;">Name</th> <th style="width: 25%;">Type</th> <th style="width: 15%;">Attribute Name</th> <th style="width: 10%;">Shared</th> <th style="width: 20%;">Report Statistics</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td>Celula 1.Queue ▾</td> <td>First In First Out</td> <td>Attribute 1</td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> <tr> <td style="text-align: center;">2</td> <td>Celula 2.Queue</td> <td>First In First Out</td> <td>Attribute 1</td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> <tr> <td style="text-align: center;">3</td> <td>Celula 3.Queue</td> <td>First In First Out</td> <td>Attribute 1</td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> <tr> <td style="text-align: center;">4</td> <td>Hold 3.Queue</td> <td>Lowest Attribute Value</td> <td>Tiempo_entrega</td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> <tr> <td style="text-align: center;">5</td> <td>Hold 4.Queue</td> <td>Lowest Attribute Value</td> <td>Tiempo_entrega</td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> <tr> <td style="text-align: center;">6</td> <td>Hold 6.Queue</td> <td>Lowest Attribute Value</td> <td>Tiempo_entrega</td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> </tbody> </table> <p style="text-align: center; color: #800000; font-weight: bold;">Ilustración 5-8 Menú opciones del Módulo Queue</p>			Name	Type	Attribute Name	Shared	Report Statistics	1	Celula 1.Queue ▾	First In First Out	Attribute 1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	2	Celula 2.Queue	First In First Out	Attribute 1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	3	Celula 3.Queue	First In First Out	Attribute 1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	4	Hold 3.Queue	Lowest Attribute Value	Tiempo_entrega	<input type="checkbox"/>	<input checked="" type="checkbox"/>	5	Hold 4.Queue	Lowest Attribute Value	Tiempo_entrega	<input type="checkbox"/>	<input checked="" type="checkbox"/>	6	Hold 6.Queue	Lowest Attribute Value	Tiempo_entrega	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	Name	Type	Attribute Name	Shared	Report Statistics																																						
1	Celula 1.Queue ▾	First In First Out	Attribute 1	<input type="checkbox"/>	<input checked="" type="checkbox"/>																																						
2	Celula 2.Queue	First In First Out	Attribute 1	<input type="checkbox"/>	<input checked="" type="checkbox"/>																																						
3	Celula 3.Queue	First In First Out	Attribute 1	<input type="checkbox"/>	<input checked="" type="checkbox"/>																																						
4	Hold 3.Queue	Lowest Attribute Value	Tiempo_entrega	<input type="checkbox"/>	<input checked="" type="checkbox"/>																																						
5	Hold 4.Queue	Lowest Attribute Value	Tiempo_entrega	<input type="checkbox"/>	<input checked="" type="checkbox"/>																																						
6	Hold 6.Queue	Lowest Attribute Value	Tiempo_entrega	<input type="checkbox"/>	<input checked="" type="checkbox"/>																																						

<i>Módulo Resource</i>	 <b>Resource</b>																																								
<p>Este módulo de datos define los recursos en un sistema de simulación, incluyendo información de costes y disponibilidad del recurso. Los recursos pueden tener una capacidad fija que no varía durante la simulación o pueden operar basándose en una planificación. Los fallos y estados del recurso se pueden definir también en este módulo.</p> <p><i>Posibles usos</i></p> <ul style="list-style-type: none"> <li>• Equipamiento (maquinaria, caja registradora, línea de teléfono).</li> <li>• Gente (empleados, procesadores de órdenes, empleados de ventas, operadores).</li> </ul> <p><i>Parámetros</i></p> <ul style="list-style-type: none"> <li>• Name: nombre del recurso cuyas características se deben definir.</li> <li>• Type: método para determinar la capacidad de un recurso. Fixed Capacity no cambiarán durante la simulación. Based on Schedule significa que se usa el módulo Schedule para especificar la capacidad y duración del recurso.</li> <li>• Capacity: número de unidades de recurso de un determinado nombre que están disponibles en el sistema para el procesamiento.</li> <li>• Schedule: name Identifica el nombre de la planificación a usar por parte del recurso. El planificador define la capacidad del recurso para un periodo de tiempo determinado.</li> <li>• Schedule Rule: determina cuándo debe ocurrir el cambio de capacidad cuando se requiere una disminución de la capacidad para una unidad de recurso muy ocupada.</li> <li>• Busy/Hour: coste por hora de un recurso que está procesando una entidad.</li> <li>• Idle/Hour: coste por hora del recurso cuando está libre.</li> <li>• Per Use: coste de un recurso en base al uso, independientemente del tiempo durante el cual se esté usando.</li> <li>• StateSet Name: nombre de los estados que se le pueden asignar a un recurso durante la simulación.</li> <li>• Initial State: estado inicial del recurso.</li> <li>• Failures: lista todos los fallos asociados con el recurso.</li> </ul>																																									
<table border="1" style="width: 100%; border-collapse: collapse; border: 2px solid red;"> <thead> <tr> <th style="width: 5%;">#</th> <th style="width: 15%;">Name</th> <th style="width: 15%;">Type</th> <th style="width: 5%;">Capacity</th> <th style="width: 5%;">Busy / Hour</th> <th style="width: 5%;">Idle / Hour</th> <th style="width: 5%;">Per Use</th> <th style="width: 10%;">StateSet Name</th> <th style="width: 5%;">Failures</th> <th style="width: 5%;">Report Statistic</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td>Maquina 1 ▾</td> <td>Fixed Capacity</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0.0</td> <td style="text-align: center;">0.0</td> <td style="text-align: center;">0.0</td> <td></td> <td style="text-align: center;">0 rows</td> <td style="text-align: center;">✓</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Maquina 2</td> <td>Fixed Capacity</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0.0</td> <td style="text-align: center;">0.0</td> <td style="text-align: center;">0.0</td> <td></td> <td style="text-align: center;">0 rows</td> <td style="text-align: center;">✓</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Maquina 3</td> <td>Fixed Capacity</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0.0</td> <td style="text-align: center;">0.0</td> <td style="text-align: center;">0.0</td> <td></td> <td style="text-align: center;">0 rows</td> <td style="text-align: center;">✓</td> </tr> </tbody> </table> <p style="text-align: center; color: red; font-weight: bold; margin-top: 5px;">Ilustración 5-9 Menú opciones del Módulo Resource</p>		#	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistic	1	Maquina 1 ▾	Fixed Capacity	1	0.0	0.0	0.0		0 rows	✓	2	Maquina 2	Fixed Capacity	1	0.0	0.0	0.0		0 rows	✓	3	Maquina 3	Fixed Capacity	1	0.0	0.0	0.0		0 rows	✓
#	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistic																																
1	Maquina 1 ▾	Fixed Capacity	1	0.0	0.0	0.0		0 rows	✓																																
2	Maquina 2	Fixed Capacity	1	0.0	0.0	0.0		0 rows	✓																																
3	Maquina 3	Fixed Capacity	1	0.0	0.0	0.0		0 rows	✓																																

**Módulo Set**

Set

Este módulo de datos define varios tipos de conjuntos, incluyendo recursos, contadores, cuentas, tipos de entidad, y figuras de entidad. Los conjuntos de recursos se pueden usar en los módulos Process (Seize, Release, Enter y Leave en el panel Advanced Transfer). Los conjuntos counter y tally se pueden usar en el módulo Record. Los conjuntos queue se pueden utilizar con Seize, Hold, Access, Request, Leave, y Allocate de los paneles Advanced Process y Advanced Transfer.

**Posibles usos**

- Máquinas que pueden realizar las mismas operaciones en un servicio de fabricación.
- Supervisores, empleados de caja en un comercio.
- Conjunto de figuras correspondientes a un conjunto de tipos de entidades.

**Parámetros**

- Name: nombre del conjunto que se va a definir.
- Type: tipo de conjunto que se va a definir.
- Members: grupo repetido que especifica los miembros del recurso en el conjunto. El orden es importante si se emplean reglas de selección del tipo Preferred Order y Cyclical.
- Resource Name: nombre del recurso a ser incluido en el conjunto de recursos.
- Tally Name: nombre de la cuenta dentro del conjunto de cuentas.
- Counter Name: nombre del contador dentro del conjunto de contadores.
- Entity Type: nombre del tipo de entidad dentro del conjunto de tipos de entidad.
- Picture Name: nombre de la imagen dentro del conjunto de imágenes.

	Name	Type	Members
1	Dibujos ▼	Entity Picture	3 rows
2	Tipos	Entity Type	3 rows

Ilustración 5-10 Menú opciones del Módulo Set

<i>Módulo Variable</i>	 Variable
------------------------	---

Este módulo de datos se utiliza para definir una dimensión de la variable y su valor(es) inicial(es). Las variables se pueden referenciar en otros módulos, se les puede reasignar un valor nuevo y se pueden emplear en cualquier expresión.

*Posibles usos*

- Número de documentos procesados por hora.
- Número serie a asignar a partes para una identificación única.
- Espacio disponible en un servidor.

*Parámetros*

- Name: nombre de la variable que se va a definir.
- Rows: número de filas en una variable con dimensión.
- Columns: número de columnas en una variable con dimensión.
- Statistics: caja de selección para determinar si se recogerán o no estadísticas
- Clear Option: define el tiempo, en el caso de ser requerido, en que el valor(es) de la variable, se reinicia al valor(es) inicial(es) especificado. Si se escoge Statistics implica reiniciar esta variable a su valor inicial siempre que las estadísticas se borren. Si se escoge System se reinicia la variable a su valor inicial siempre que se reinicia el sistema. None indica que nunca se reinicia la variable a su valor inicial.
- Initial Values: lista el valor(es) inicial de la variable.
- Initial Value: valor variable al inicio de la simulación.

	Name	Rows	Columns	Clear Option	Initial Values	Report Statistics
1	Tiempo_transporte ▾			System	1 rows	<input type="checkbox"/>
2	Num_Tarjetas1			System	1 rows	<input type="checkbox"/>
3	Num_Tarjetas2			System	1 rows	<input type="checkbox"/>
4	Num_Tarjetas3			System	1 rows	<input type="checkbox"/>

**Ilustración 5-11 Menú opciones del Módulo Variable**

## Módulos de flujo. Advanced Process

<i>Módulo Signal</i>	
<p>El módulo de señal o módulo “Signal” envía un valor de señal a cada módulo de retención (Hold) en el modelo que figura en “Wait for Signal” y libera la cantidad máxima especificada de las entidades.</p> <p>Cuando una entidad llega a un módulo de señal, se evalúa la señal y el código de señal se envía. En este momento, las entidades que están en espera en el módulo Hold de la misma señal son liberadas de las colas del citado módulo. La entidad que envía la señal a su paso por el módulo continuará hasta que encuentra un retardo, entre en una cola, o en un módulo “disposed”.</p> <p><i>Posibles usos</i></p> <ul style="list-style-type: none"> <li>• Análisis de los patrones de tráfico en una intersección (señal cuando la luz cambie a verde)</li> <li>• Señalización de un operador para completar una orden que estaba esperando un componente</li> </ul> <p><i>Parámetros</i></p> <ul style="list-style-type: none"> <li>• Name: identificador único del módulo que se muestra en la forma del módulo.</li> <li>• Signal Value: valor de la señal que se enviará a las entidades en los módulos espera (Hold).</li> <li>• Limit: número máximo de entidades que van a ser liberados de cualquier módulo Hold cuando se recibe la señal</li> </ul>	
<div data-bbox="528 1167 1050 1503" data-label="Image"> </div> <p data-bbox="539 1509 1043 1536"><b>Ilustración 5-12 Menú opciones del Módulo Signal</b></p>	

<i>Módulo Hold</i>	 <b>Hold</b>
<p>Este módulo retendrá una entidad en una cola para esperar a una señal o que una condición llegue a ser verdadera (Scan) o sea detenida infinitamente, para que sea removida después con el módulo Remove.</p> <p>Si la entidad está detenida esperando una señal, el módulo Signal se utiliza en otro lugar en el modelo para permitir que la entidad pase al siguiente módulo. Si la entidad está esperando que una condición dada sea verdadera, la entidad permanecerá en el módulo, hasta que la condición/es llegue a ser verdadera.</p> <p>Cuando la entidad es un Hold infinito, el módulo Remove se utiliza en algún lugar del modelo para permitir que la entidad continúe procesándose.</p> <p><i>Posibles usos</i></p> <ul style="list-style-type: none"> <li>• Espera a que un semáforo se vuelva verde.</li> <li>• Deteniendo una pieza esperando una autorización.</li> <li>• Comprobando el estado de una máquina u operador para continuar un proceso.</li> </ul> <p><i>Parámetros</i></p> <ul style="list-style-type: none"> <li>• Name: identificador único del módulo que se muestra en la forma del módulo.</li> <li>• Type: indica el razonamiento de espera de la entidad en una cola interna o especificada. Wait for Signal, esperará la entidad hasta que se reciba una señal del mismo valor. Scan for Condition esperará la entidad hasta que una condición específica sea verdadera. Con la opción Infinite Hold, la entidad esperará hasta que sea retirada de la cola por el módulo Remove.</li> <li>• Wait for Value: código de señal para la entidad en espera. Se aplica solo cuando Type es Wait for Signal.</li> <li>• Limit: número máximo de entidades en espera que serán liberadas una vez recibida una señal. Se aplica solo cuando Type es Wait for a Signal.</li> <li>• Condition: especifica la condición que será evaluada para que espere la entidad en el módulo. Si la condición evaluada es verdadera, la entidad deja el módulo inmediatamente. Si la condición es falsa, la entidad esperará en la cola asociada hasta que la condición se vuelva verdadera. Se aplica solo cuando Type es Scan for Condition.</li> <li>• Queue Type: determina el tipo de cola en el que esperan las entidades. Si se selecciona Queue, se especifica el nombre de la cola. Si se selecciona Set, se especifican la cola seleccionada y los miembros. Si se selecciona Internal, se emplea una cola interna para la espera de todas las entidades. Attribute y Expression son métodos adicionales para definir la utilización de la cola.</li> <li>• Queue Name: es visible si Queue Type es Queue y define el nombre simbólico de la cola.</li> <li>• Set Name: este campo solo es visible si Queue Type is Set y define la cola seleccionada que contiene la cola que se referencia.</li> </ul>	

- Set Index: este campo es visible solo si Queue Type es Set y define una referencia en la cola seleccionada.
- Attribute: este campo es visible solo si Queue Type es Attribute. El atributo introducido en este campo será evaluado para indicar que cola se va a utilizar.
- Expression: este campo solo es visible su Queue Type es expression. La expresión introducida en este campo será evaluada para indicar que cola se emplea.

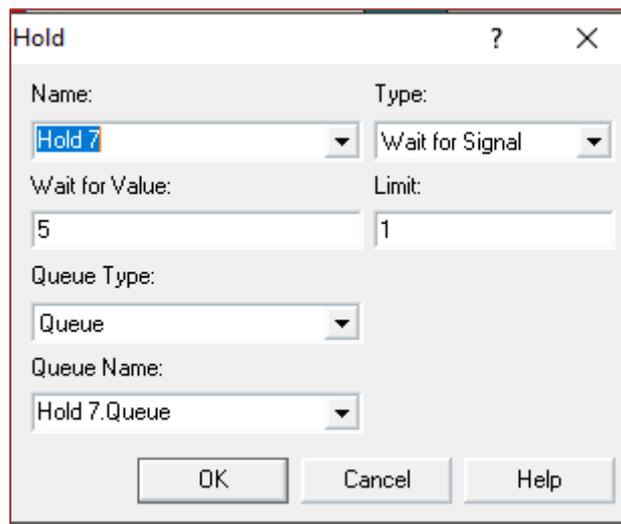
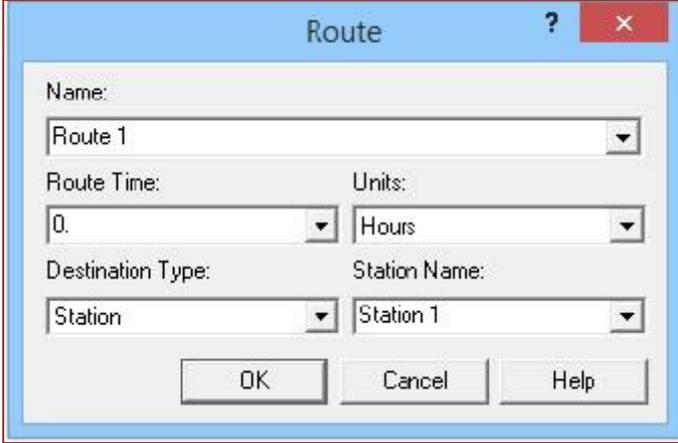


Ilustración 5-13 Menú opciones del Módulo Hold

## Módulos de flujo. Advanced Transfer

Módulo Route	
<p>Route: Este módulo transfiere una entidad a una estación especificada o a la siguiente estación definida por una secuencia de visitas para la entidad. Se puede definir un tiempo de retardo para transferir a la siguiente estación.</p> <p><i>Parámetros:</i></p> <ul style="list-style-type: none"> <li>• Name: Nombre que se le da para identificar al módulo, debe ser único.</li> <li>• Route Time: Tiempo de viaje desde la ubicación actual de la entidad a la estación de destino.</li> <li>• Units: Unidades de tiempo para el parámetro Route Time.</li> <li>• Destination type: Método para determinar el destino de la entidad. Se puede seleccionar Station, Sequential, Attribute y Expression. Si es por secuencia, se requiere que a la entidad se le haya asignado una secuencia y que esa secuencia haya sido definida. Para las simulaciones realizadas se hace uso de Station y Sequential.</li> <li>• Station name: Nombre de la estación de destino. En caso de elegir Station.</li> </ul>	
 <p style="text-align: center;"><b>Ilustración 5-14 Menú opciones del Módulo Route</b></p>	

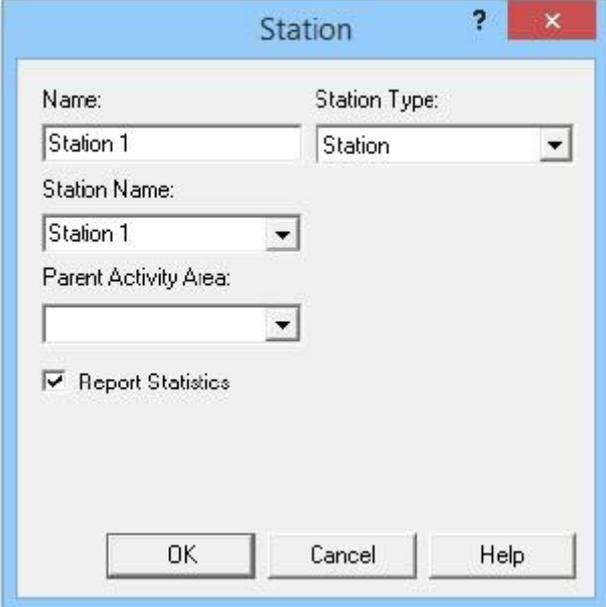
**Módulo Station**

Station 1

Station: Este módulo define una ubicación donde se produce el procesamiento. Se usa por la claridad que le da al modelo al separar ciertos procesos y hace más fácil su manejo.

**Parámetros:**

- Name: Nombre que se le da para identificar al módulo, debe ser único.
- Station Type: Tipo de estación que se define, puede ser individual o un conjunto, para el caso en el que se centra este proyecto solo se definen individuales.
- Station Name: Nombre de la estación individual.
- Parent Activity Area: Nombre del área de actividad.
- Report Statistics: Casilla de verificación que especifica si las estadísticas se recopilarán automáticamente y se almacenarán en la base de datos del informe para esta estación y su área de actividad correspondiente.



**Ilustración 5-15 Menú opciones del Módulo Station**

## Capítulo 6: Modelado de los sistemas Kanban.

### Sentido del Modelo

Este capítulo se centra en la descripción del Modelo de Arena Simulation utilizado para realizar la simulación sistema de control Kanban.

Dentro de las simulaciones que serán realizadas, serán estudiados los casos siguientes:

- Entorno con Baja Variabilidad en Llegada de pedidos y Baja Saturación en colas
- Entorno con Alta Variabilidad en Llegada de pedidos y Alta Saturación en colas
- Entorno con Baja Variabilidad en Llegada de pedidos y Alta Saturación en colas
- Entorno con Alta Variabilidad en Llegada de pedidos y Baja Saturación en colas

### Módulos

En la siguiente ilustración, se verá el modelo completo que será utilizado para simular el entorno controlado con sistema Kanban y posteriormente veremos un desglose de cada una de las partes del mismo y una explicación de sus módulos y funcionamientos.

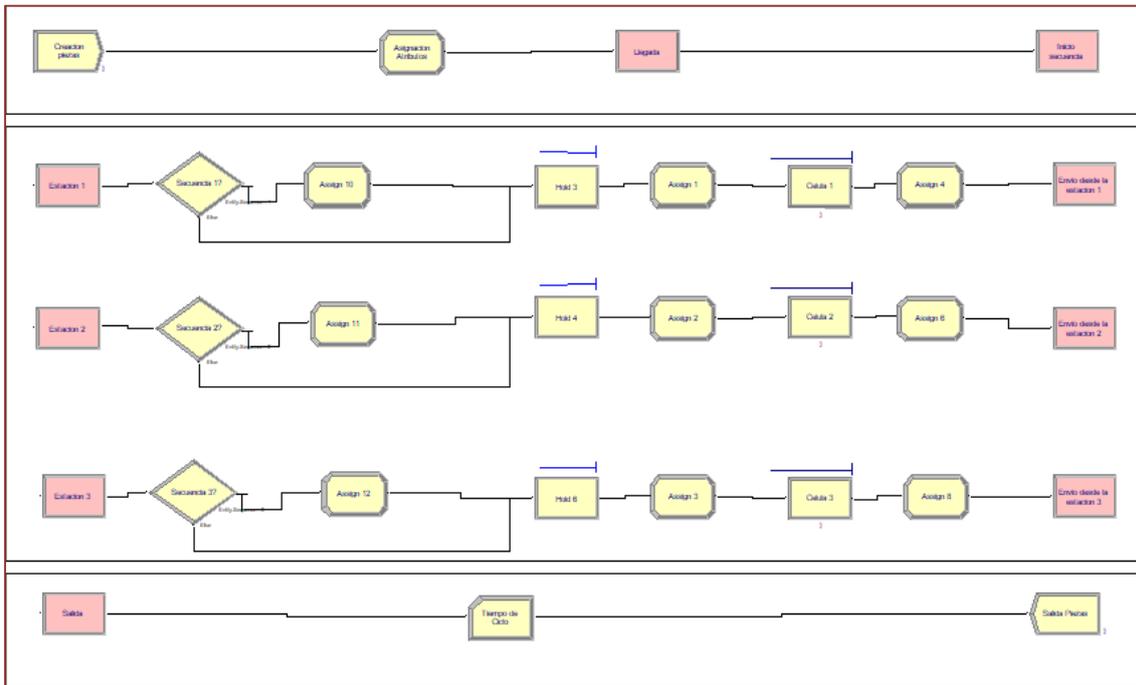


Ilustración 6-1. Modelo Completo de Kanban

## Modulo 1

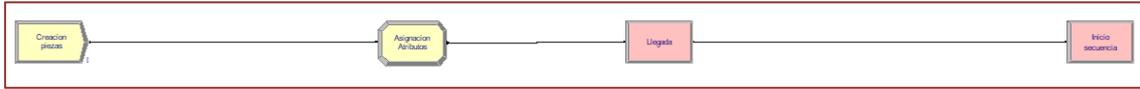


Ilustración 6-2. Modulo de entrada

El Módulo 1 se corresponde con las condiciones de entradas. En él se encuentra en primer lugar un Módulo Create (Creación de piezas), para la creación de las entidades tipo pedidos. Seguido se observa un Módulo Assign (Asignación Atributos) en el que se le asigna a cada pedido unos atributos determinados. Posteriormente, se tiene la Estación de Llegada, para luego dar comienzo a la secuencia previamente asignada según el tipo de producto que se esté fabricando, en nuestro caso:

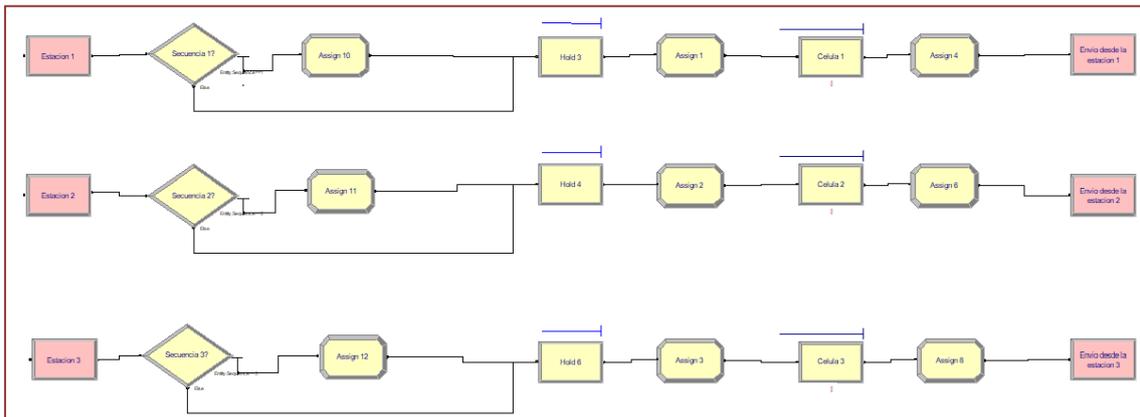
Bicicleta tipo 1	
Bicicleta tipo 2	
Bicicleta tipo 3	

Tabla 6-1. Tipos producto

El Módulo Assign se encargará de asignarle a cada entidad que por él pase, unos atributos determinados, entre los que encontramos:

- Tiempo de entrega
- Tipo
- Secuencia
- Pictures

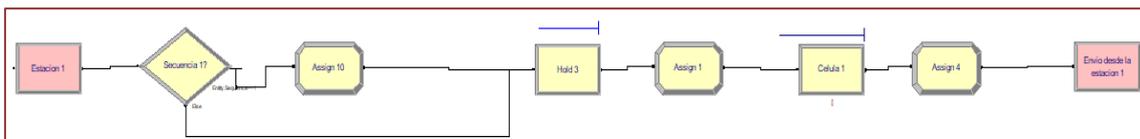
## Modulo 2



**Ilustración 6-3. Módulo de procesado**

El módulo 2 corresponde con la zona de trabajo. En esta se puede observar tres estaciones en las que se realizan unas operaciones determinadas.

En la siguiente imagen se muestra una de las tres líneas y se explican los módulos que contienen y su función.



**Ilustración 6-4. Línea de producción en estación 1**

- Módulo Station: Es el módulo al que se dirige la entidad según su ruta.
- Modulo Decide: Tiene la función de comprobar si es la primera máquina por la que pasa la entidad para así determinar fácilmente el tiempo de ciclo
- Módulo Hold: Su función no es más que la de retener cada pieza hasta el momento en el que haya al menos una tarjeta Kanban disponible. Una vez que se libere una tarjeta Kanban, se le asignara a la próxima pieza en cola, lo que permitirá que esta comience a ser procesada.
- Módulo Assign: Es aquí donde se realiza la asignación de la tarjeta Kanban para iniciar el proceso. Este módulo funciona además como contador, en este caso, su función es disminuir el número de tarjetas Kanban disponibles en una unidad en el momento que una entidad pasa por él.
- Módulo Process: Es el modulo en el que se procesa la entidad.
- Módulo Assign: Al igual que el anterior, este es un contador de tarjetas Kanban. En el momento que una entidad lo atraviesa, el número de tarjetas Kanban disponibles incrementa en una unidad.
- Módulo Route: Es el modulo utilizado para dirigir a la entidad hacia la próxima estación en la que le corresponde ser procesada según su ruta.

### Módulo 3

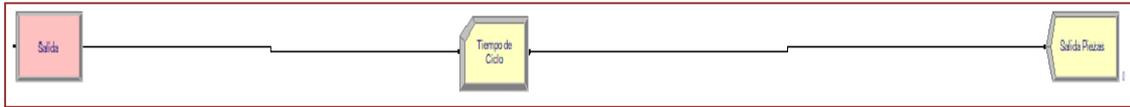


Ilustración 6-5. Módulo de Salida

El Modulo 3 de la planta de producción se corresponde con la salida del producto. Una vez que la entidad ha pasado por todas las etapas anteriores, pasa a la fase de salida, donde el producto abandona la línea.

El modelo cuenta con un Módulo Record, en él se obtiene el tiempo salida para así calcular el tiempo que la entidad pasa en el sistema, es decir, el tiempo de ciclo.

### Tablero de control



Ilustración 6-6. Tablero de control

En el modelo, esto es una representación aproximada de lo que el controlador de tarjetas vería. Cada sección representa una estación. Aquí se puede ver rápidamente el número de tarjetas Kanbans disponibles en cada estación y el WIP de las máquinas de la factoría. A partir de aquí, el controlador podrá liberar pedidos para ser procesados en función de la disponibilidad de tarjetas, únicamente.

## Capítulo 7: Modelado de los sistemas Cobacabana.

### Sentido del Modelo

Este capítulo se centra en la descripción del Modelo de Arena Simulation utilizado para realizar la simulación sistema de control Kanban.

Dentro de las simulaciones que serán realizadas, serán estudiados los casos siguientes:

- Entorno con Baja Variabilidad en Llegada de pedidos y Baja Saturación en colas
- Entorno con Alta Variabilidad en Llegada de pedidos y Alta Saturación en colas
- Entorno con Baja Variabilidad en Llegada de pedidos y Alta Saturación en colas
- Entorno con Alta Variabilidad en Llegada de pedidos y Baja Saturación en colas

### Módulos

En la siguiente ilustración, se verá el modelo completo que será utilizado para simular el entorno controlado con sistema Cobacabana y posteriormente veremos un desglose de cada una de las partes del mismo y una explicación de sus módulos y funcionamientos.

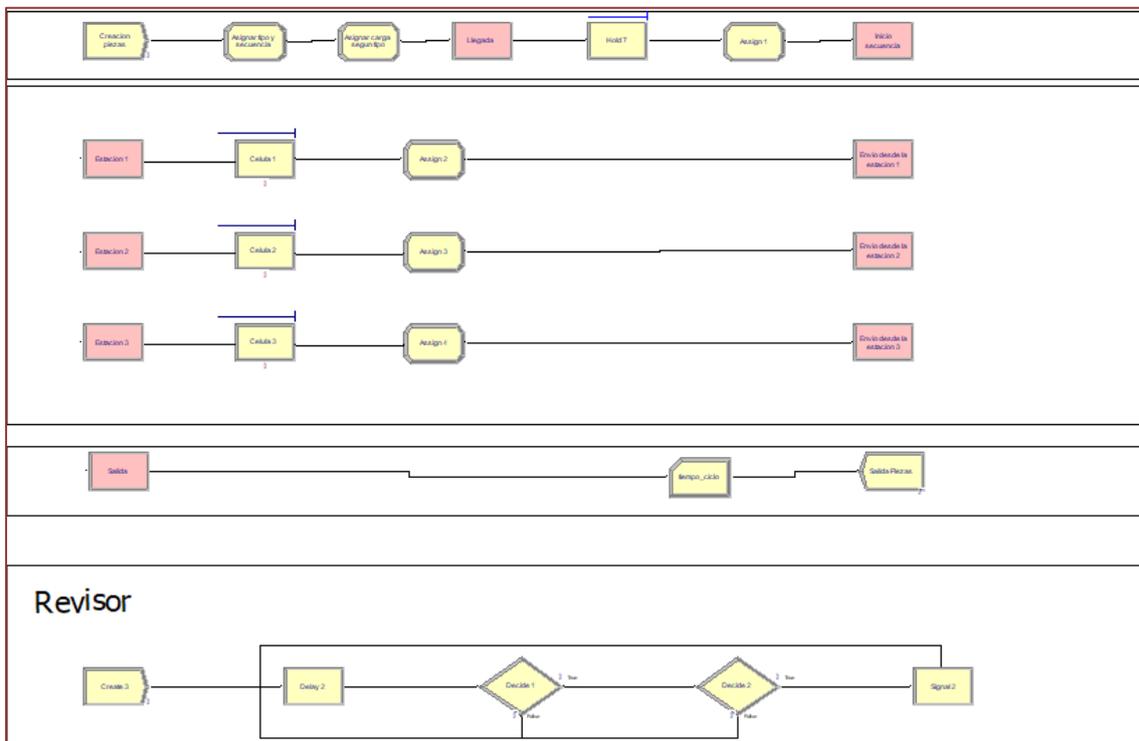


Ilustración 7-1 Modelo Completo de Cobacabana

## Módulo 1

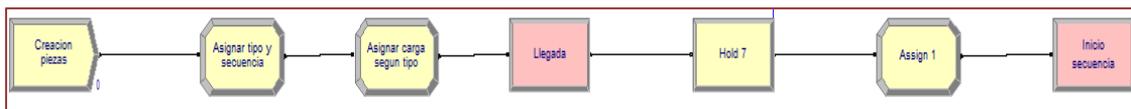


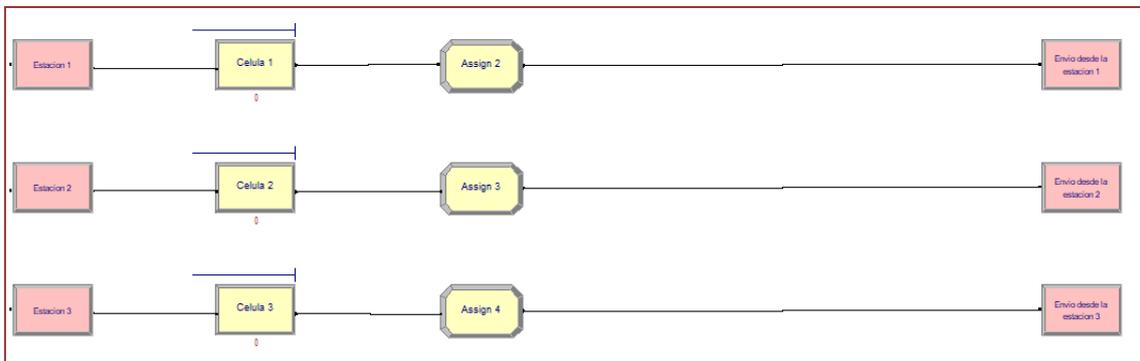
Ilustración 7-2. Modulo de entrada

El Módulo 1 se corresponde con las condiciones de entradas. En él se encuentra en primer lugar un Módulo Create (Creación de piezas), para la creación de las entidades tipo pedidos. Seguido se observa un Módulo Assign (Asignación Atributos) en el que se le asigna a cada pedido unos atributos determinados. Posteriormente, encontramos otro Modulo Assign en el que, según el tipo de pedido, se le asigna al mismo unas cargas determinadas para cada estación. Luego se tiene la Estación de Llegada, a la que le sigue un Módulo Hold que representa el Pull de órdenes, aquí se esperará hasta que el valor de la carga del próximo pedido para cada estación por los que pasará, permita que este se fabrique sin ninguna interrupción. Cabe destacar que el orden de procesamiento es según la regla de despacho EDD (Early Due Date). Una vez que las estaciones tienen un nivel de carga que permita que la entidad se procesa, el Hold la libera y pasa por un Módulo Assign, que le asignara a cada entidad un tiempo de entrada en el sistema, el cual utilizaremos para calcular el tiempo el producto pasa en el sistema, es decir el tiempo de ciclo. Luego la entidad pasa a un Módulo Route dar comienzo a la secuencia previamente asignada según el tipo de producto que se esté fabricando, en nuestro caso:

Bicicleta tipo 1	
Bicicleta tipo 2	
Bicicleta tipo 3	

Tabla 7-1. Tipos producto

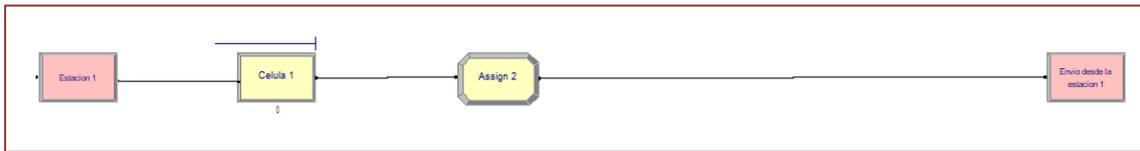
## Módulo 2



**Ilustración 7-3. Módulo de procesado**

El módulo 2 corresponde con la zona de trabajo. En esta se puede observar tres estaciones en las que se realizan unas operaciones determinadas.

En la siguiente imagen se muestra una de las tres líneas y se explican los módulos que contienen y su función.



**Ilustración 7-4. Línea de producción en estación 1**

- Módulo Station: Es el módulo al que se dirige la entidad según su ruta.
- Módulo Hold: Su función no es más que la de retener cada pieza hasta el momento en el que haya al menos una tarjeta Kanban disponible. Una vez que se libere una tarjeta Kanban, se le asignará a la próxima pieza en cola, lo que permitirá que esta comience a ser procesada.
- Módulo Assign: Es aquí donde se realiza la asignación de la tarjeta Kanban para iniciar el proceso. Este módulo funciona además como contador, en este caso, su función es disminuir el número de tarjetas Kanban disponibles en una unidad en el momento que una entidad pasa por él.
- Módulo Process: Es el módulo en el que se procesa la entidad.
- Módulo Assign: Al igual que el anterior, este es un contador de tarjetas Kanban. En el momento que una entidad lo atraviesa, el número de tarjetas Kanban disponibles incrementa en una unidad.
- Módulo Route: Es el módulo utilizado para dirigir a la entidad hacia la próxima estación en la que le corresponde ser procesada según su ruta.

### Módulo 3



Ilustración 7-5. Módulo de Salida

El Módulo 3 de la planta de producción se corresponde con la salida del producto. Una vez que la entidad ha pasado por todas las etapas anteriores, pasa a la fase de salida, donde el producto abandona la línea.

El modelo cuenta con un Módulo Record, en él se obtiene el tiempo salida para así calcular el tiempo que la entidad pasa en el sistema, es decir, el tiempo de ciclo.

### Revisor

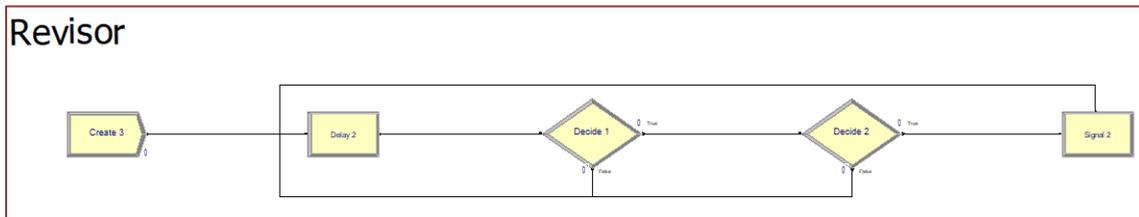


Ilustración 7-6 Modulo demonio, simulador de revisor

Este Módulo es el que simula al revisor, la persona que se encarga de liberar los pedidos para que sean procesados. Consta de un Módulo Create, que es el que crea la entidad revisor. Seguido de este se encuentra un Módulo Delay, cuya función es la de evitar que el revisor este mirando el nivel de carga constantemente, sino que se establece un tiempo ocioso, por así decirlo, en el que el revisor no está observando el display de cargas. Posteriormente aparecen dos Módulos Decide que se encargaran de establecer las condiciones para liberar una entidad en el Pool de Ordenes para posteriormente ser procesadas. Las condiciones que mira el sistema de control Cobacabana es que la carga de trabajo que implicaría fabricar esta entidad en las estaciones, sumada con el nivel de carga actual, no supera el 100%.

## Control de Cargas



Ilustración 7-7 Tablero de control de cargas

En el modelo, esto es una representación aproximada de lo que el controlador de cargas vería. Cada sección representa una estación. Aquí se puede ver rápidamente el nivel de carga que posee cada estación. A partir de aquí, el controlador podrá liberar pedidos para ser procesados en función de la carga disponible en cada estación.

## Capítulo 8: Discusión de Resultados.

En este capítulo se van a analizar los resultados obtenidos mediante las simulaciones realizadas. En primer lugar, un estudio comparativo de los tres sistemas de control de la producción basados en tarjetas que se han explicado y, en segundo lugar, las propuestas de solución ante el bloqueo cruzado.

### Estudio comparativo

Para realizar el estudio comparativo de los sistemas de control de la producción detallados anteriormente, Kanban y Cobacabana se hace uso de cuatro indicadores y tres factores.

Los tres indicadores son:

- Tiempo de ciclo promedio total: es el tiempo desde que la orden de trabajo empieza el proceso hasta que termina.
- WIP promedio: número de unidades que se están procesando, promediadas a lo largo del tiempo de simulación.
- Utilización: tiempo que está la máquina ocupada dividido entre el tiempo que está disponible, es decir, porcentaje de uso que se le da a ésta.

Para esto, como se mencionó anteriormente, se simularon cuatro escenarios distintos, para estudiar el comportamiento de un entorno controlado por el sistema Kanban y el comportamiento del mismo controlado por Cobacabana.

### Escenarios

Escenario	Congestion	Variabilidad
Tipo 1	Alta Congestión	Baja Variabilidad
Tipo 2	Alta Congestión	Alta Variabilidad
Tipo 3	Baja Congestión	Baja Variabilidad
Tipo 4	Baja Congestión	Alta Variabilidad

Tabla 8-1. Tipos escenarios

Los el valor de los factores y parámetros utilizados en para la simulación son los siguientes:

Se tendrán dos factores y la combinación entre ellos, da lugar a 4 escenarios distintos. Los factores que se emplean son:

- Variabilidad: se estudia el comportamiento de los sistemas ante la baja y la alta variabilidad de los tiempos de proceso de las máquinas.

- Congestión: se estudia el comportamiento de los sistemas ante la baja y alta aglomeración de las órdenes de trabajo en el sistema.
- Para conseguir una alta congestión se han establecido unos tiempos de llegada según una normal con media 5 y desviación 0.5 [NORM (5, 0.5)] con este factor, conseguimos una utilización superior al 90% en el entorno controlado por Kanban, este mismo será usado en el entorno controlado por Cobacabana para poder comparar los resultados.
- Para conseguir una baja congestión se han establecido unos tiempos de llegada según una normal con media 5 y desviación 0.5 [NORM (6.5, 0.5)] con este factor, conseguimos una utilización superior al 80% pero inferior al 90% en el entorno controlado por Kanban, este mismo será usado en el entorno controlado por Cobacabana para poder comparar los resultados.
- Para conseguir una alta variabilidad se han establecido unos tiempos de proceso que siguen una distribución normal con media 7 y desviación 3.5 [NORM (7, 3.5)].
- Para conseguir una baja variabilidad se han establecido unos tiempos de proceso que siguen una distribución normal con media 7 y desviación 0.2 [NORM (7, 0.2)].

## Resultados Completos

En este apartado se detalla cada uno de los escenarios para cada tipo de sistema de control. Además se incluyen los factores y parámetros utilizados y los resultados obtenidos tras la simulación del escenario. Posteriormente, se encuentra una tabla resumen que facilitará la lectura de estos resultados y ayudara a obtener unas conclusiones.

### Alta congestión y Alta variabilidad.

#### Kanban

Tiempos de llegada: NORM (5, 0.5)

Tiempos de Proceso: NORM (7,3.5)

#### Tiempo de Ciclo

Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
Tiempo de Ciclo	67.9759	5.28	46.8841	86.8004	2.0067	1479.54

Ilustración 8-1Tiempo de ciclo Kanban Escenario tipo 1

Datos para calcular **WIP promedio**

Number Busy



Ilustración 8-1 Number Busy Kanban escenario tipo 1

Number in Queue

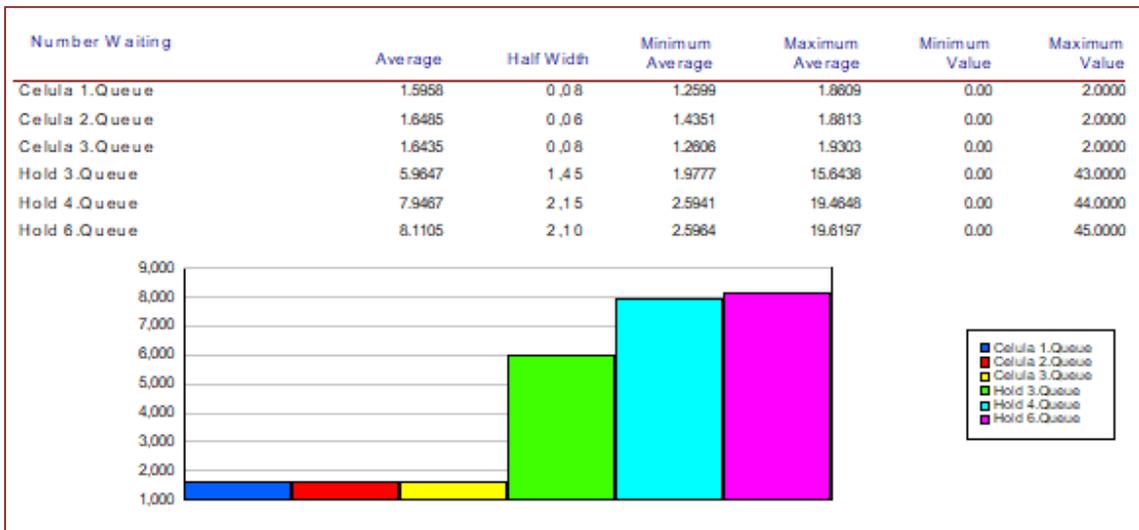


Ilustración 8-2 Number in Queue Kanban escenario tipo 1

## Discusión de Resultados.

### Cálculos:

Number Busy		Hold.Queue		Number in Queue		Suma
Maquina 1	0,92	Hold 3	11,4048	Célula 1	1,595	13,9198
Maquina 2	0,9269	Hold 4	7,6118	Célula 2	1,648	10,1867
Maquina 3	0,9275	Hold 5	9,8687	Célula 3	1,643	12,4392
					WIP	36,5457

Tabla 8-2. Calculos WIP Kanban escenario tipo 1

### Utilización

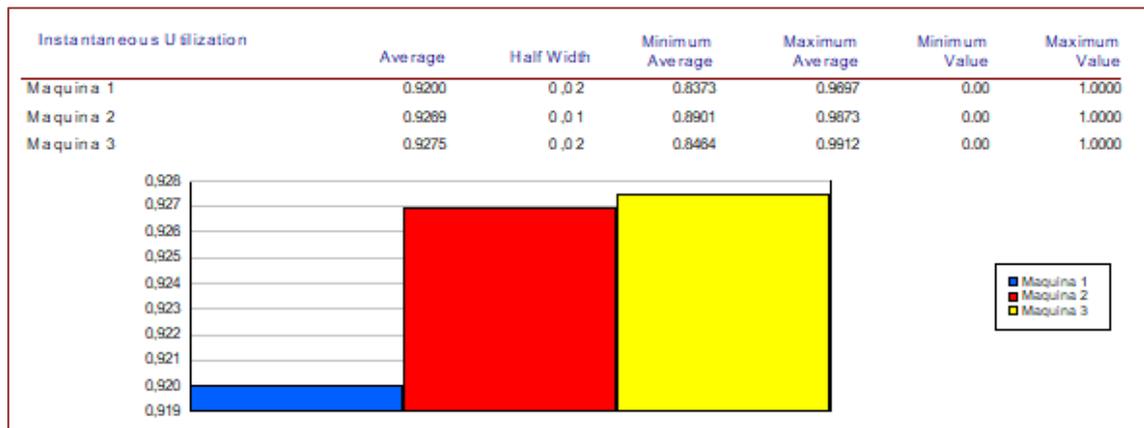


Ilustración 8-3 Utilización Kanban escenario tipo 1

### Resultados Kanban

Utilización	Valores
Maquina 1	0.9200
Maquina 2	0.9269
Maquina 3	0.9275
WIP promedio	36,5457
Tiempo de Ciclo promedio	67,9759

Tabla 8-3. Tabla resumen resultados Kanban escenario tipo 1

## Cobacabana

Tiempos de Llegada: NORM (5, 0.5)

Tiempos de Proceso: NORM (7,3.5)

### Tiempo de Ciclo

Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
tiempo_ciclo	37.9474	0,28	36.6941	39.0042	11.6243	77.9497

Ilustración 8-5 Tiempo de ciclo Cobacabana Escenario tipo 1

Datos para calcular **WIP promedio**

### Number Busy

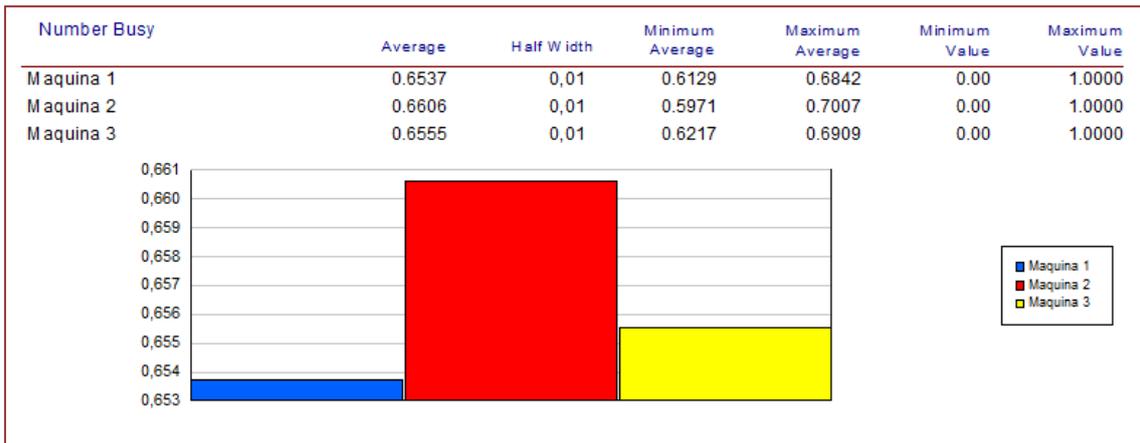


Ilustración 8-6 Number Busy Cobacabana escenario tipo 1

### Number in Queue

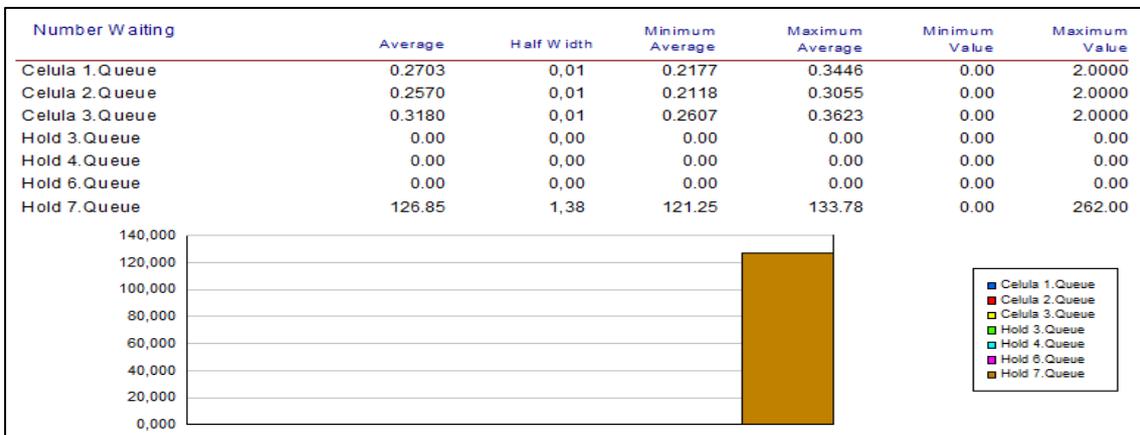


Ilustración 8-7 Number in Queue Cobacabana escenario tipo 1

## Discusión de Resultados.

Cálculos:

Numeber Busy		Number in Queue		Suma
Maquina 1	0,6537	Celula 1	0,2177	0,8714
Maquina 2	0,6606	Celula 2	0,2094	0,87
Maquina 3	0,6555	Celula 3	0,235	0,8905
			WIP	2,6319

Tabla 8-4. Calculos WIP Cobacabana escenario tipo 1

### Utilización

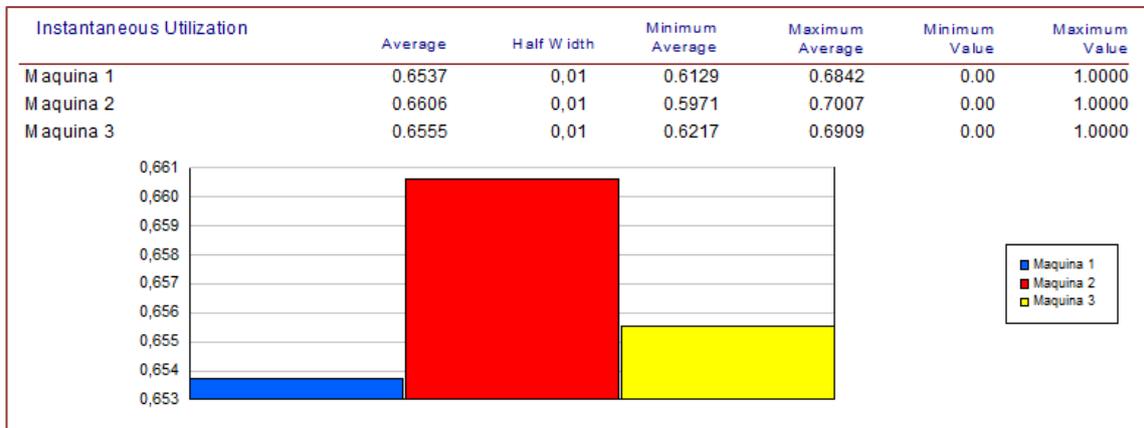


Ilustración 8-8 Utilización Cobacabana escenario tipo 1

### Resultados Cobacabana

Utilización	Valores
Maquina 1	0,6537
Maquina 2	0,6606
Maquina 3	0,6555
WIP promedio	2,6319
Tiempo de Ciclo promedio	37,9474

Tabla 8-5. Tabla resumen resultados Cobacabana escenario tipo 1

## Alta congestión y Baja variabilidad

### Kanban

Tiempos de Llegada: NORM (5, 0.5)

Tiempos de Proceso: NORM (7,0.2)

### Tiempo de Ciclo

Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
Tiempo de Ciclo	61.9854	5.29	33.2275	82.3842	2.0025	2101.83

Ilustración 8-9 Tiempo de ciclo Kanban Escenario tipo 2

### Datos para calcular **WIP promedio**

#### Number Busy

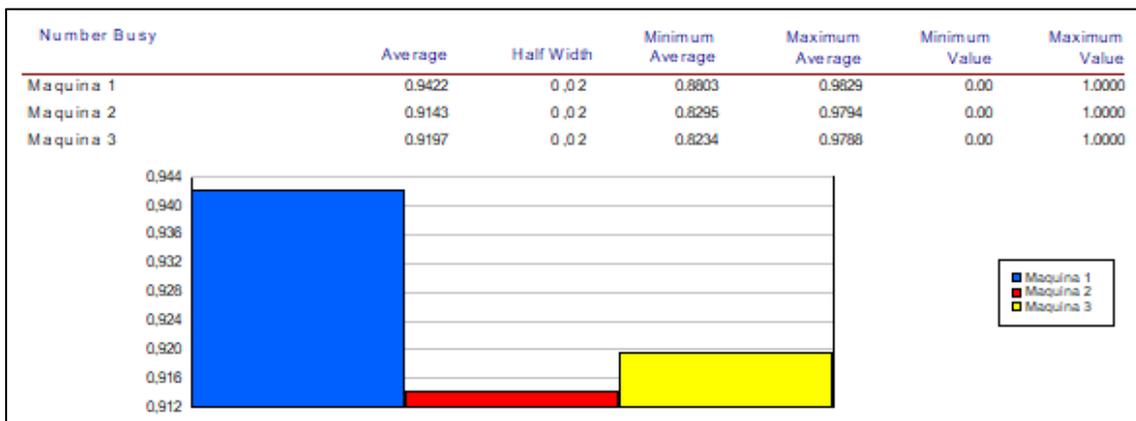


Ilustración 8-10 Number Busy Kanban escenario tipo 2

#### Number in Queue

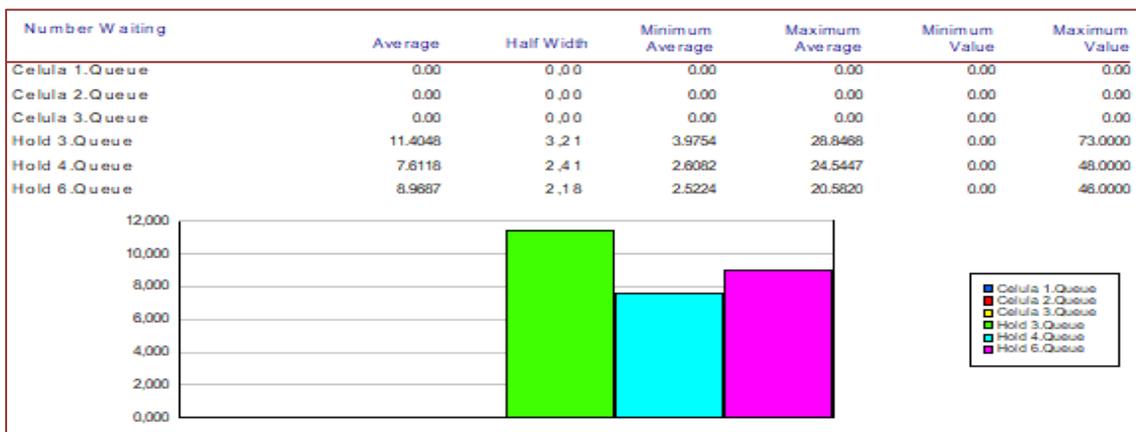


Ilustración 8-11 Number in Queue Kanban escenario tipo 2

## Discusión de Resultados.

Cálculos:

Number Busy		Hold.Queue		Number in Queue		Suma
Maquina 1	0,9422	Hold 3	11,4048	Célula 1	0	12,347
Maquina 2	0,9143	Hold 4	7,6118	Célula 2	0	8,5261
Maquina 3	0,9197	Hold 5	9,8687	Célula 3	0	10,7884
WIP						31,6615

**Tabla 8-7. Calculos WIP Kanban escenario tipo 2**

**Utilización**



**Ilustración 8-12 Utilización Kanban escenario tipo 2**

Resultados Kanban Escenario 2

Utilización	Valores
Maquina 1	0,9422
Maquina 2	0,9143
Maquina 3	0,9197
WIP promedio	31,6615
Tiempo de Ciclo promedio	61,984

**Tabla 8-8. Tabla resumen resultados Kanban escenario tipo 2**

## Cobacabana

Tiempos de llegada: NORM (5, 0.5)

Tiempos de Proceso: NORM (7,0.2)

**Tiempo de Ciclo**

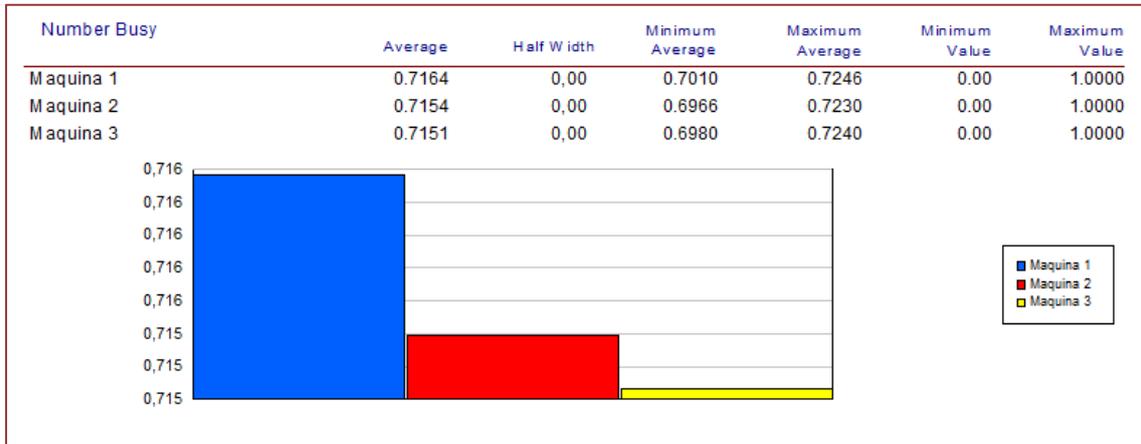
Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
tiempo_ciclo	35.6228	0.11	35.2940	36.0424	27.9663	60.8861

**Ilustración 8-13 Tiempo de ciclo Cobacabana Escenario tipo 2**

## Discusión de Resultados.

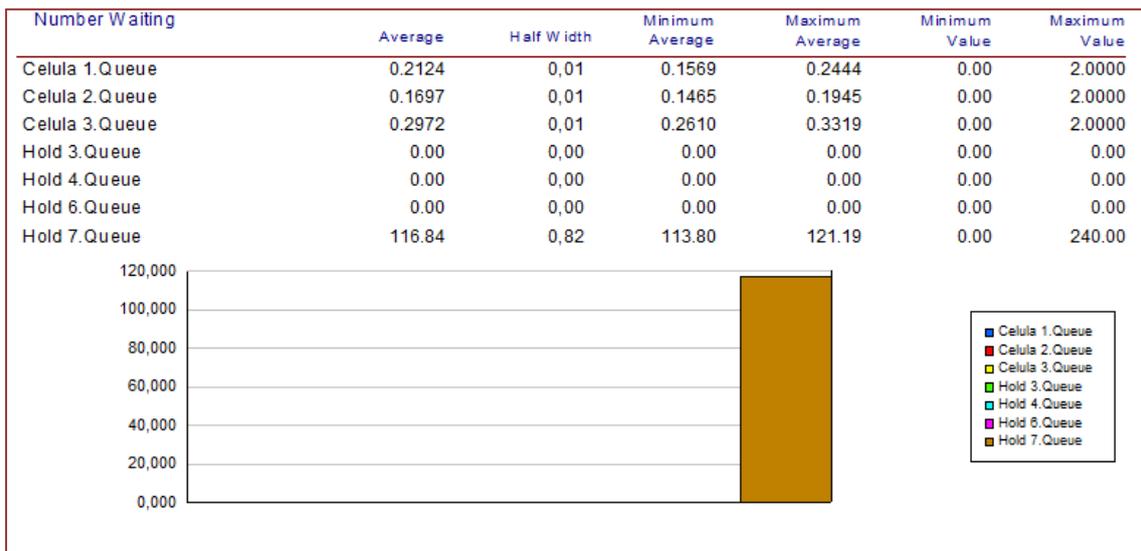
### Datos para calcular **WIP promedio**

#### Number Busy



**Ilustración 8-14 Number Busy Cobacabana escenario tipo 2**

#### Number in Queue



**Ilustración 8-15 Number in Queue Cobacabana escenario tipo 2**

#### Cálculos:

Numeber Busy		Number in Queue		Suma
Maquina 1	0,7164	Celula 1	0,2124	0,9288
Maquina 2	0,7154	Celula 2	0,1697	0,8851
Maquina 3	0,7151	Celula 3	0,2972	0,924
			<b>WIP</b>	<b>2,7379</b>

**Tabla 8-9. Calculos WIP Cobacabana escenario tipo 2**

### Utilización

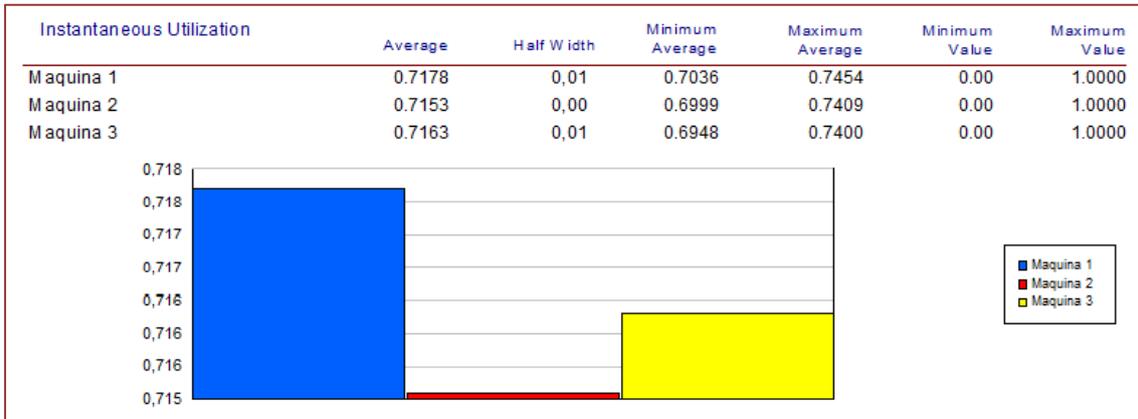


Ilustración 8-16 Utilización Cobacabana escenario tipo 2

### Resultados Cobacabana

Utilización	Valores
Maquina 1	0,7164
Maquina 2	0,7154
Maquina 3	0,7151
WIP promedio	2,7379
Tiempo de Ciclo promedio	35,6228

Tabla 8-10. Tabla resumen resultados Cobacabana escenario tipo 2

## Baja congestión y Alta variabilidad

### Kanban

Tiempos de llegada: NORM (6.5, 0.5)

Tiempos de Proceso: NORM (7,3.5)

### Tiempo de Ciclo

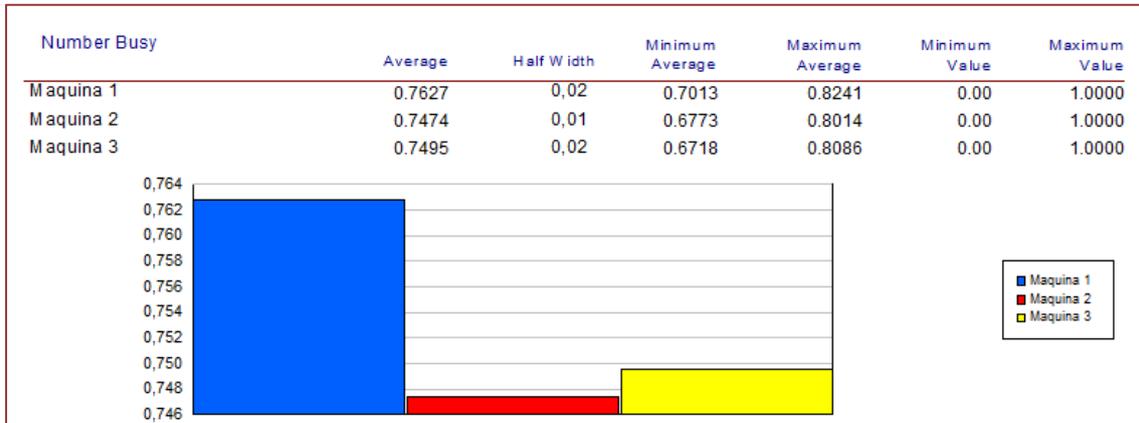
Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
Tiempo de Ciclo	27.5204	2,31	21.3294	40.1847	2.0017	817.18

Ilustración 8-27 Tiempo de ciclo Kanban Escenario tipo 3

## Discusión de Resultados.

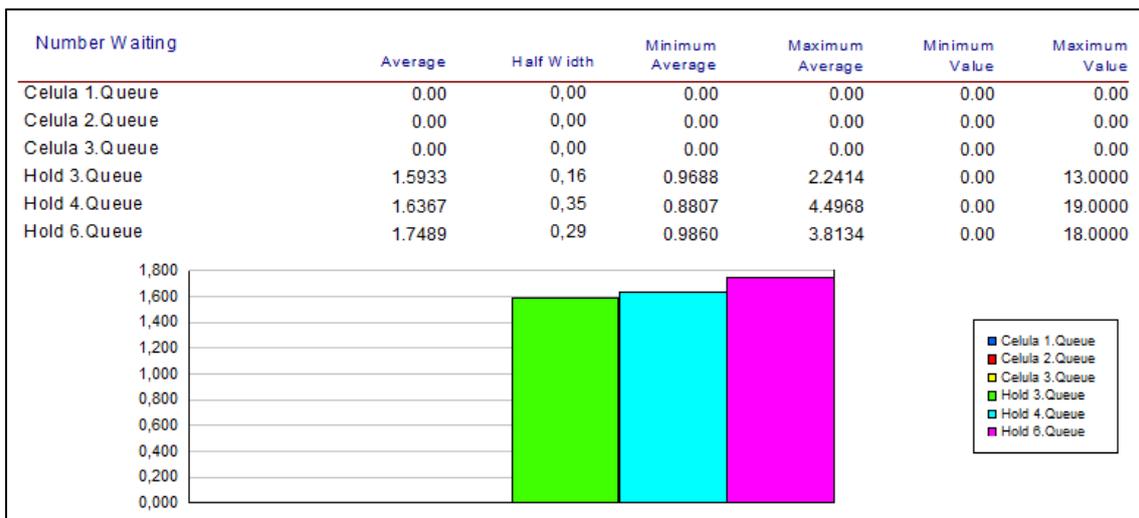
### Datos para calcular **WIP promedio**

#### Number Busy



**Ilustración 8-18 Number Busy Kanban escenario tipo 3**

#### Number in Queue



**Ilustración 8-19 Number in Queue Kanban escenario tipo 3**

#### Cálculos:

Number Busy		Hold.Queue		Number in Queue		Suma
Maquina 1	0,7627	Hold 3	1,5933	Célula 1	0	2,356
Maquina 2	0,7474	Hold 4	1,6367	Célula 2	0	2,3841
Maquina 3	0,7495	Hold 5	1,7489	Célula 3	0	2,4984
					<b>WIP</b>	<b>7,2385</b>

**Tabla 8-11. Calculos WIP Kanban escenario tipo 3**

### Utilización

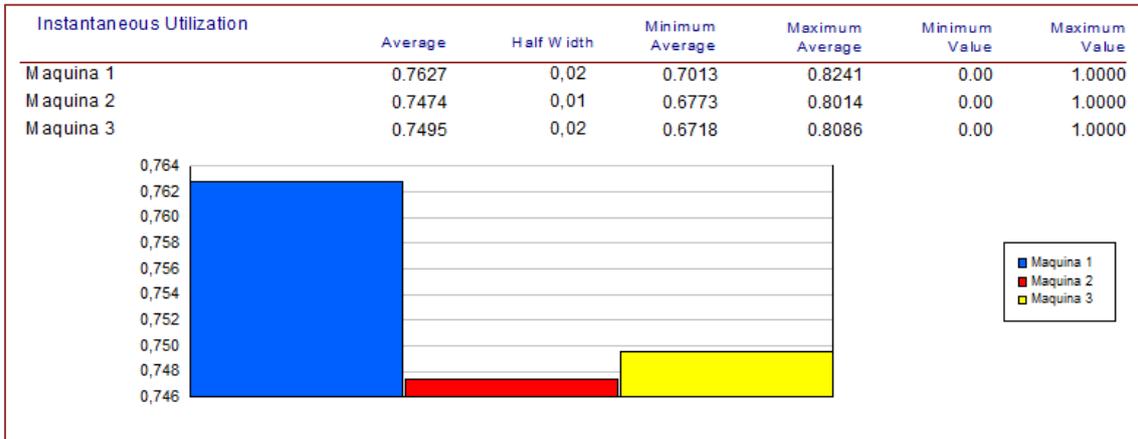


Ilustración 8-20 Utilización Kanban escenario tipo 3

### Resultados Kanban Escenario 3

Utilización	Valores
Maquina 1	0,7627
Maquina 2	0,7474
Maquina 3	0,7495
WIP promedio	7,238
Tiempo de Ciclo promedio	27,2504

Tabla 8-12. Tabla resumen resultados Kanban escenario tipo 3

### Cobacabana

Tiempos de Llegada: NORM (6.5, 0.5)

Tiempos de Proceso: NORM (7,3.5)

### Tiempo de Ciclo

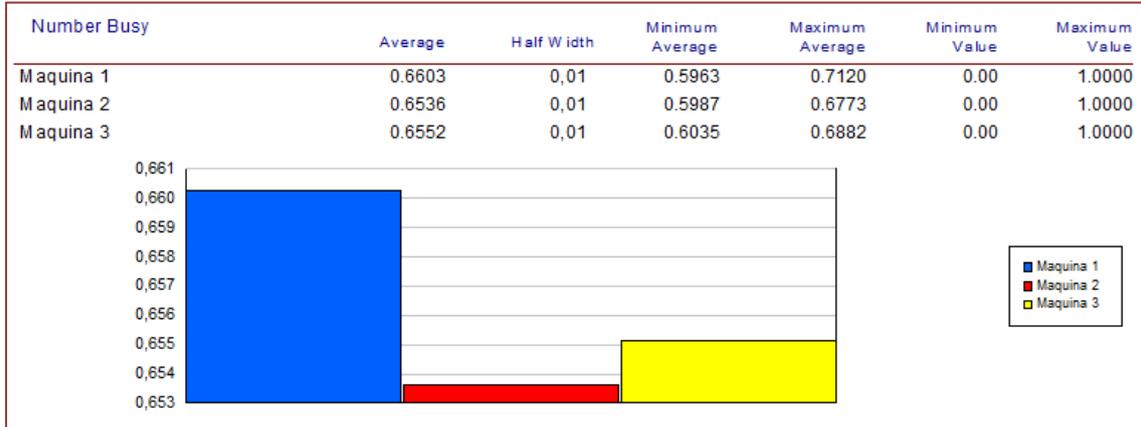
Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
tiempo_ciclo	38.1903	0,35	36.8361	39.6569	14.5577	79.7544

Ilustración 8-23 Tiempo de ciclo Cobacabana Escenario tipo 3

## Discusión de Resultados.

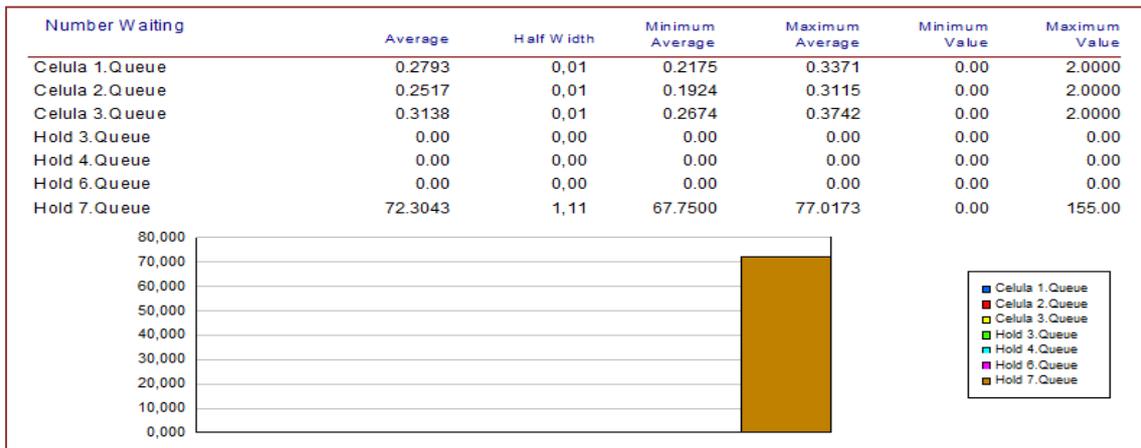
### Datos para calcular **WIP promedio**

#### Number Busy



**Ilustración 8-24 Number Busy Cobacabana escenario tipo 3**

#### Number in Queue



**Ilustración 8-25 Number in Queue Cobacabana escenario tipo 3**

#### Cálculos:

Numeber Busy		Number in Queue		Suma
Maquina 1	0,6603	Celula 1	0,2793	0,9396
Maquina 2	0,6536	Celula 2	0,2517	0,9053
Maquina 3	0,6552	Celula 3	0,3138	0,969
			WIP	2,8139

**Tabla 8-13. Calculos WIP Cobacabana escenario tipo 3**

**Utilización**

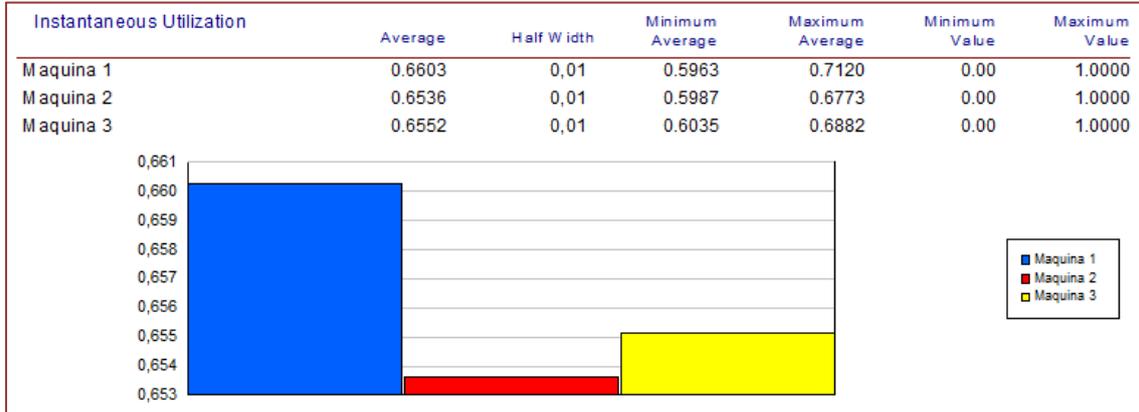


Ilustración 8-26 Utilización Cobacabana escenario tipo 3

**Resultados Cobacabana**

Utilización	Valores
Maquina 1	0,6603
Maquina 2	0,6536
Maquina 3	0,6552
WIP promedio	2,8139
Tiempo de Ciclo promedio	38,1903

Tabla 8-14. Tabla resumen resultados Cobacabana escenario tipo 3

**Baja congestión y Baja variabilidad**

**Kanban**

Tiempos de llegada: NORM (6.5, 0.5)

Tiempos de Proceso: NORM (7,0.2)

**Tiempo de Ciclo**

Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
Tiempo de Ciclo	27.5204	2,31	21.3294	40.1847	2.0017	817.18

Ilustración 8-25 Tiempo de ciclo Kanban Escenario tipo 4

## Discusión de Resultados.

### Datos para calcular WIP promedio

#### Number Busy

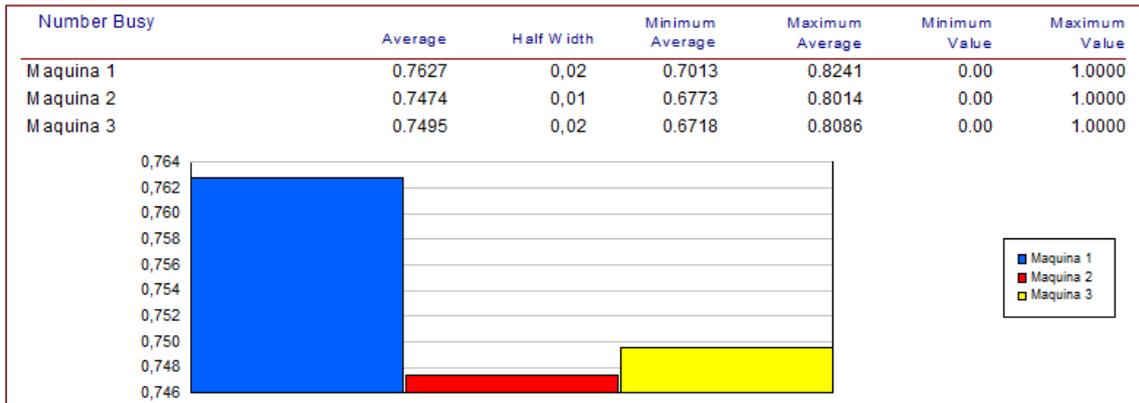


Ilustración 8-76 Number Busy Kanban escenario tipo 4

#### Number in Queue

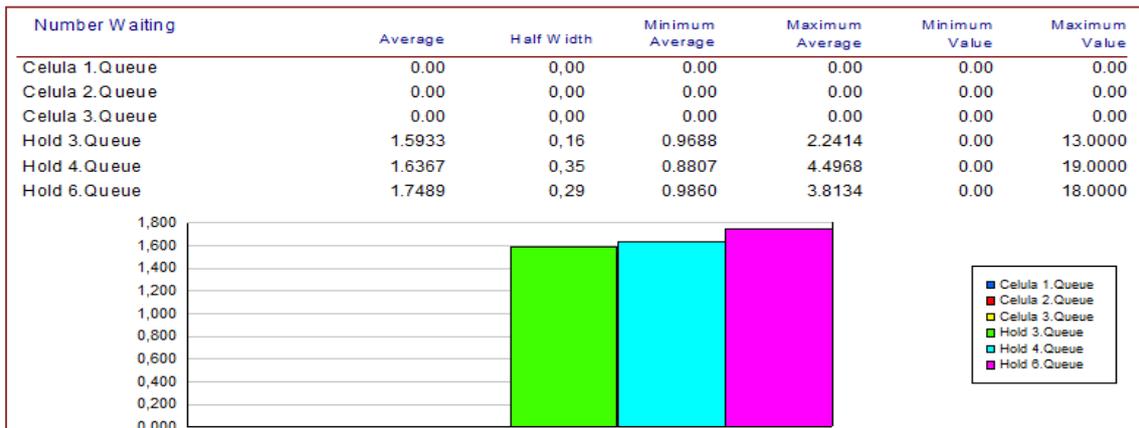


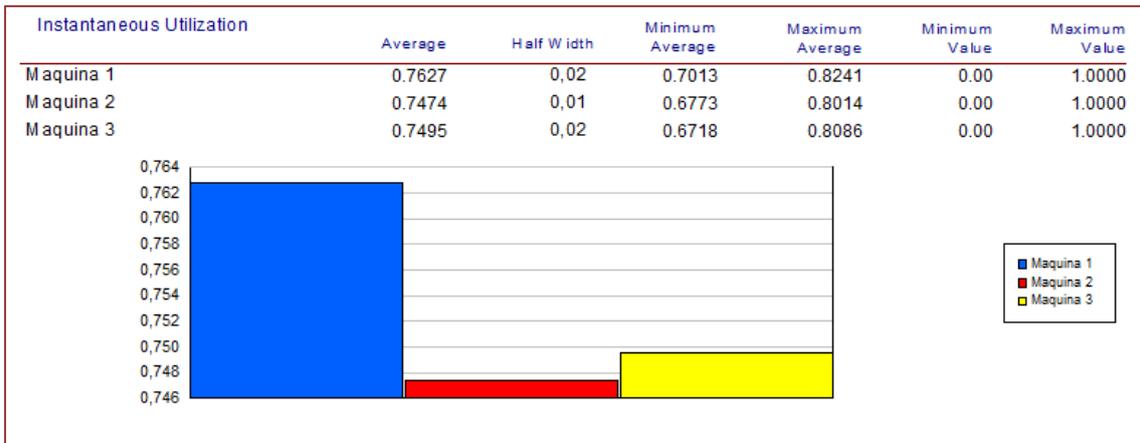
Ilustración 8-27 Number in Queue Kanban escenario tipo 4

#### Cálculos:

Number Busy		Hold.Queue		Number in Queue		Suma	
Maquina 1	0,7627	Hold 3	1,5933	Célula 1	0	2,356	
Maquina 2	0,7474	Hold 4	1,6367	Célula 2	0	2,3841	
Maquina 3	0,7495	Hold 5	1,7489	Célula 3	0	2,4984	
						WIP	7,2385

Tabla 8-15. Calculos WIP Kanban escenario tipo 4

**Utilización**



**Ilustración 8-28 Utilización Kanban escenario tipo 4**

**Resultados Kanban**

Utilización	Valores
Maquina 1	0,7627
Maquina 2	0,7474
Maquina 3	0,7495
WIP promedio	7,238
Tiempo de Ciclo promedio	27,2504

**Tabla 8-16. Tabla resumen resultados Kanban escenario tipo 4**

**Cobacabana**

Tiempos de llegada: NORM (6.5, 0.5)

Tiempos de Proceso: NORM (7,0.2)

**Tiempo de Ciclo**

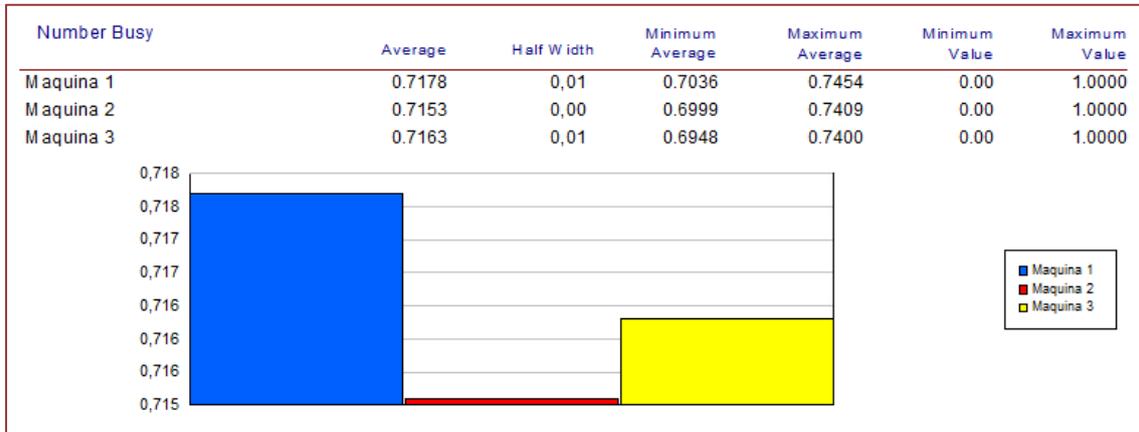
Expression	Average	Half Width	Minimum Average	Maximum Average	Minimum Value	Maximum Value
tiempo_ciclo	35.6724	0,15	35.1058	36.4536	28.0280	56.3874

**Ilustración 8-29 Tiempo de ciclo Cobacabana Escenario tipo 4**

## Discusión de Resultados.

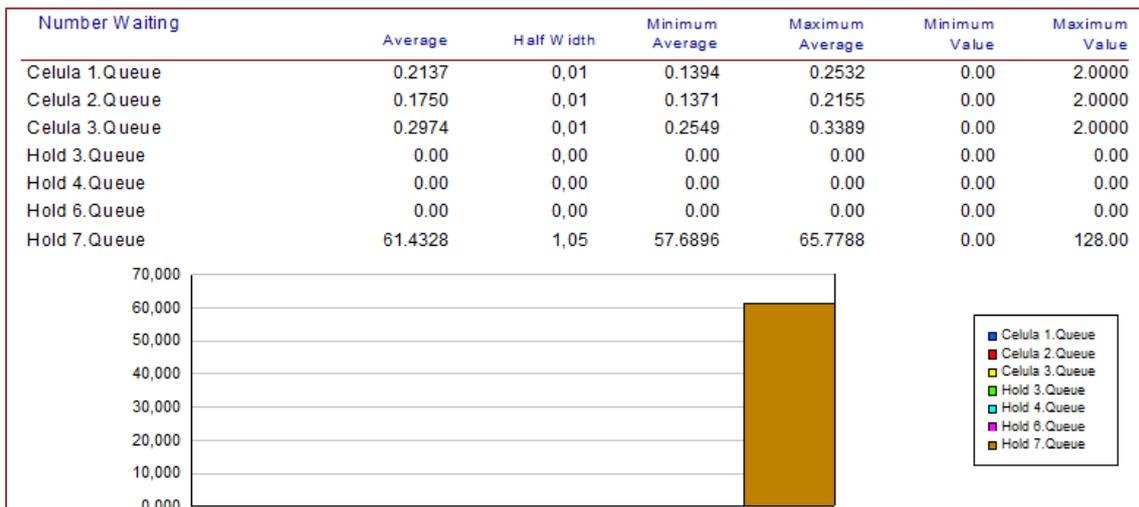
### Datos para calcular **WIP promedio**

#### Number Busy



**Ilustración 8-30 Number Busy Cobacabana escenario tipo 4**

#### Number in Queue



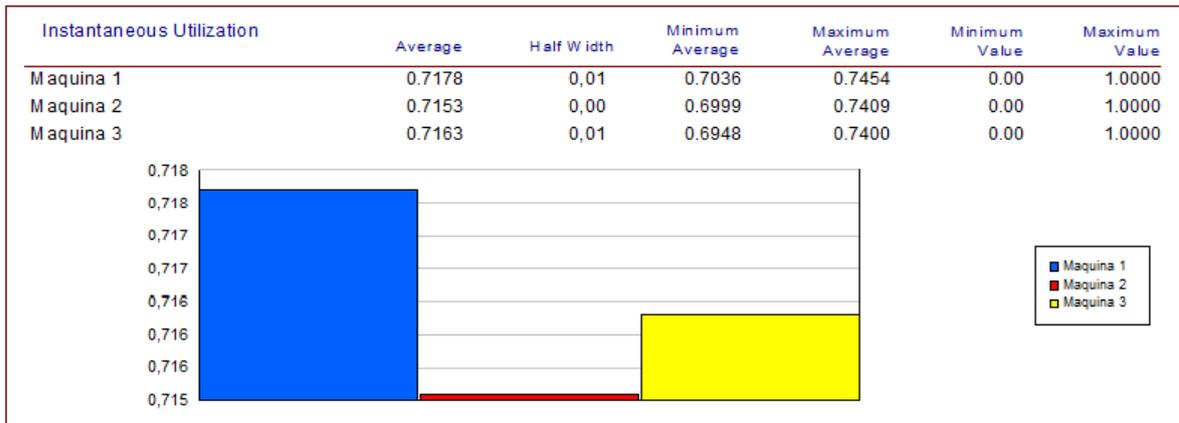
**Ilustración 8-81 Number in Queue Cobacabana escenario tipo 4**

#### Cálculos:

Numeber Busy		Number in Queue		Suma
Maquina 1	0,7178	Celula 1	0,2137	0,9315
Maquina 2	0,7153	Celula 2	0,175	0,8903
Maquina 3	0,7163	Celula 3	0,2974	1,0137
			WIP	2,8355

**Tabla 8-17. Calculos WIP Cobacabana escenario tipo 4**

**Utilización**



**Ilustración 8-32 Utilización Cobacabana escenario tipo 4**

**Resultados Cobacabana**

Utilización	Valores
Maquina 1	0,7178
Maquina 2	0,7153
Maquina 3	0,7163
WIP promedio	2,8355
Tiempo de Ciclo promedio	35,6724

**Tabla 8-18. Tabla resumen resultados Cobacabanaescenario tipo 4**

**Tabla con resultados**

A continuación se mostrara una tabla en la que se podran observar todos los resultados obtenidos en las distintas simulaciones realizadas. De esta forma se podra realizar una comparacion mas rapida de los dos entornos ya que se pueden ver los datos facilmente, sin necesidad de buscar en tablas distintas, labor que, aunque no compleja, puede conllevar a errores de lectura.

Sistema	Indicadores	Tipo 1	Tipo 2	Tipo 3	Tipo 4	
Kanban	Tiempo de ciclo	67,9759	61,984	27,2504	27,2504	
	WIP promedio	36,5457	31,6615	7,2385	7,2385	
	Utilización	Máquina 1	0,92	0,9422	0,7627	0,7627
		Máquina 2	0,9269	0,9143	0,7474	0,7474
Máquina 3		0,9275	0,9197	0,7495	0,7495	
Cobacabana	Tiempo de ciclo	37,9474	35,6228	35,6724	38,1903	
	WIP promedio	2,6319	2,8262	2,8355	2,8139	
	Utilización	Maquina 1	0,6537	0,7164	0,7178	0,6603
		Maquina 2	0,6606	0,7154	0,7153	0,6536
Máquina 3		0,6555	0,7151	0,7163	0,6552	

**Tabla 8-19. Tabla resumen resultados**

Leyenda Escenarios	
Tipo 1	Alta congestión-Alta variabilidad
Tipo 2	Alta congestión-Baja variabilidad
Tipo 3	Baja congestión-Baja variabilidad
Tipo 4	Baja congestión- Alta variabilidad

Tabla 8-20. Tabla Leyendas

Como se puede observar en la tabla anterior, el sistema de control Cobacabana cumple con lo teóricamente establecido, mantener un WIP reducido, en este caso, consigue superar al sistema Kanban ya que este obtiene unos valores más elevados en todos los escenarios. Ante esta situación si se quiere que el taller tenga un nivel de WIP bajo, la implementación de un sistema de control Cobacabana sería una buena opción. Ahora bien, esto se consigue debido a que este sistema no permite que entren más órdenes al proceso si éstas no pueden ser procesadas sin ninguna interrupción. Es decir, si el nivel de carga de la entidad implicaría que el porcentaje de utilización de cada máquina de cada estación sobrepase el 100%, esta entidad se esperará en el Pool de órdenes hasta que pueda ser procesada completamente. Esta es la parte negativa, por decirlo de algún modo, del sistema Cobacabana, obtenemos un nivel bajo de WIP a costa de tener una cola, que en algunos casos puede ser enorme, en el Pool de órdenes.

Por otro lado, se puede observar que para el tiempo de ciclo, Kanban supera a Cobacabana para entornos en los que la congestión es baja. Aunque exista más trabajo en proceso, los tiempos de llegada y de proceso, permiten que el sistema Kanban trabaje correctamente obteniendo un tiempo de ciclo que es inferior al que proporciona Cobacabana. En cambio, funciona peor para escenarios en los que la congestión es alta.

Finalmente, la utilización de las maquinas en Cobacabana es inferior, en todos los casos a la utilización de las maquinas con un sistema Kanban

## Capítulo 9: Conclusiones y líneas futuras

Este capítulo presenta las conclusiones extraídas del análisis de los datos obtenidos tras las simulaciones realizadas de los sistemas de control de producción presentados en este proyecto. Además de una relación de posibles líneas futuras para ampliar este estudio.

### Conclusiones

Tras las simulaciones realizadas a los dos sistemas de control de producción, Kanban y Copacabana, y comparar los resultados, se puede decir para cualquiera de los escenarios, Cobacabana nos ofrece unos mejores resultados cuando queremos reducir el WIP promedio eso sí, a costa de incrementar el número de unidades en cola que, aunque no estén en el sistema, se encuentran esperando la orden de aceptación para entrar en él.

Por otro lado, se ve como para entornos en los que la congestión es baja, Kanban nos ofrece menor tiempo de ciclo, es decir, las entidades pasan menos tiempo en el sistema, pero el WIP sigue siendo descontrolado.

### Líneas futuras

Hay varias líneas futuras interesantes para ampliar el estudio realizado en este proyecto como son:

1. En primer lugar, se encuentra el utilizar Cobacabana mejorado, aquel en el que si se tiene en cuenta la posición de la estaciones de la ruta para asignar la carga de trabajo de la entidad.
2. Realizar las pruebas en entornos de simulación que no se encuentren equilibrados.

Arena user's guide.

Champan, S.N. (2006). Planificación y control de la producción. Pearson Prentice Hall.

Cuatrecasas, L. (2011). Organización de la producción y dirección de operaciones: sistemas actuales de gestión eficiente y competitiva. Diaz de Santos.

Harrod, S., Kanet, J.J. (2013). "Applying Work Flow Control in Make-to-Order Job Shops". International Journal of Production Economics, 143: 620-626.

Heizer, J., Render, B. (2008). Dirección de la producción y operaciones: decisiones tácticas. Pearson Prentice Hall, 8ª Edición.

Hopp, W. y Spearman, M. (2011). Factory Physics. McGraw-Hill, 3ª Edición.

Hopp, W.J., Spearman M.L. (2004). "To Pull or Not to Pull: What Is the Question?". Manufacturing & Service Operations Management, 6(2):133-148.

Lödding, H. (2013). Handbook of manufacturing control: fundamentals, description, configuration. Springer.

Moscoso, P., Lago, A. (2016). Gestión de operaciones para directivos: destapa el pleno potencial de tu empresa. McGraw-Hill Education.

Ohno, T. (1988). Toyota Production System: Beyond Large Scale Production. 1st ed. Cambridge, MA: Productivity Press.

Riezebos, J. (2010). "Design of POLCA Material Control Systems". International Journal of Production Research, 48(5): 1455-1477

Suri, R. (1998). Quick Response Manufacturing: A Companywide Approach to Reducing Leadtimes. Cambridge, MA: Productivity Press.

Thürer, M., Stevenson, M., Protzman, C.W. (2016). "Card-based production control: a review of the control mechanisms underpinning Kanban, ConWIP, POLCA and COBACABANA systems". Production Planning and Control, 27(14): 1143-1157.

Vandaele, N., Nieuwenhuysse, I.V., Claerhout, D., Cremmery, R. (2008). "Load-Based POLCA: An Integrated Material Control System for Multiproduct". Multimachine Job Shops, Manufacturing & Service Operations Management, 10(2): 181-197.

Velasco, J., Campins, J.A. (2013). Gestión de la producción en la empresa: planificación, programación y control. Ediciones Pirámide

Land, M.J., 2009. Cobacabana (control of balance by card-based navigation): A card-based system for job shop control. International Journal of Production Economics, 117(1), 97–10