MASTER OF SCIENCE BY RESEARCH

Smart Factory: Developing a Digital Twin and Intelligent Network for Self-Adaptive, Flexible Production Environments

Rollinson, Simon

Award date:
2020

Awarding institution:
Coventry University

Smart Factory: Developing a Digital Twin and Intelligent Network for Self-Adaptive, Flexible Production Environments

Author: Simon Rollinson

Submission Date: May 2019

A thesis submitted in partial fulfilment of the University's requirements for the Degree of Master of Research.

Certificate of Ethical Approval

Applicant:

Simon Rollinson

Project Title:

Smart Factory: developing cyber physical systems and intelligent networks for self-adaptive, flexible production environments

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Medium Risk

Date of approval:

13 April 2018

Project Reference Number:

P61796

Abstract

The purpose of this research was to examine the feasibility of creating an autonomous closed-loop Digital Twin system based around Discrete Event Simulation in order to increase production efficiency.

The dissertation is based on a literature review of the definition and use case examples of Digital Twins in industry. This is followed by interviews with Crown process experts to ascertain the feasibility of utilising the different sources of input data required to create the Digital Twin.

The research suggests that there would be a benefit in terms of production order optimisation to be gained from the development of an autonomous Digital Twin of $15.6million globally per annum, if implemented on all lines running small batches. However the use of Discrete Event Simulation in its traditional sense is recommended for the purpose of predicting the line performance impact of capital investment projects.

In terms of Digital Twin development, it could be argued that Smartline, Crown's in house developed process monitoring software, is a visualisation Digital Twin as it monitors and stores the PLC data in real time and the data is available to view and analyse from anywhere in the world, given the required security access. Crown could enhance Smartline as a Digital Twin in four ways:

1. Include the physical layout of the lines and in particular the conveyor sizes to visualise work in progress.
2. Link to the Enterprise Resource Planning (ERP) system to enable customer order planning on Smartline
3. Migrate the Control system from the PLC computers to Smartline.
4. Develop a regional overview system in order to group lines by product which may be in different factories to perform the production planning and logistics across sites.

The enhanced visualisation of data would allow Crown to identify methods to increase production and reduce spoilage.

Crown has many factories with similar production lines across the globe and could benefit from a big data analysis project, not as a one off, but as an ongoing monitoring and optimisation system. The benefits of clustering, including fuzzy clustering, of machines globally to monitor and compare performance would allow global performance improvements, by targeting improvement projects to bring the lower performing machines closer to the best performers.

Contents

Tables & Figures

List of Tables

List of Figures

1 Introduction

1.1 Project Background

Crown Packaging currently has 143 manufacturing plants worldwide operating in highly competitive markets with very tight profit margins. Crown strives to be the world leader in product development with the aim of reduction in raw material and other operating costs.

Crown is looking to introduce Industry 4.0 technologies into factories with the goal of increasing productivity by approximately 3% in order to reveal the "Hidden Factories". An increase in production efficiency of 3% would provide the extra output of 3 to 4 factories.

Additionally, metal packaging is under pressure from competitive packaging technologies become more agile, i.e. to give a more customisable output, and offer smaller batch quantities.

The Industry 4.0 techniques would be targeted at:

- Optimise layouts
- Optimise line controls
- Optimising production order selection
- Autonomous production lines adapting to batch variables.

1.2 Crown Packaging

Crown Packaging manufacturing capabilities are broadly separated into three categories:

1. Beverage cans
2. Food cans
3. Speciality Packaging, including biscuit tins and aerosol cans.

1.2.1 Beverage

The beverage business is the largest part of the Crown portfolio and also the most advanced of the businesses in terms of technological implementation.

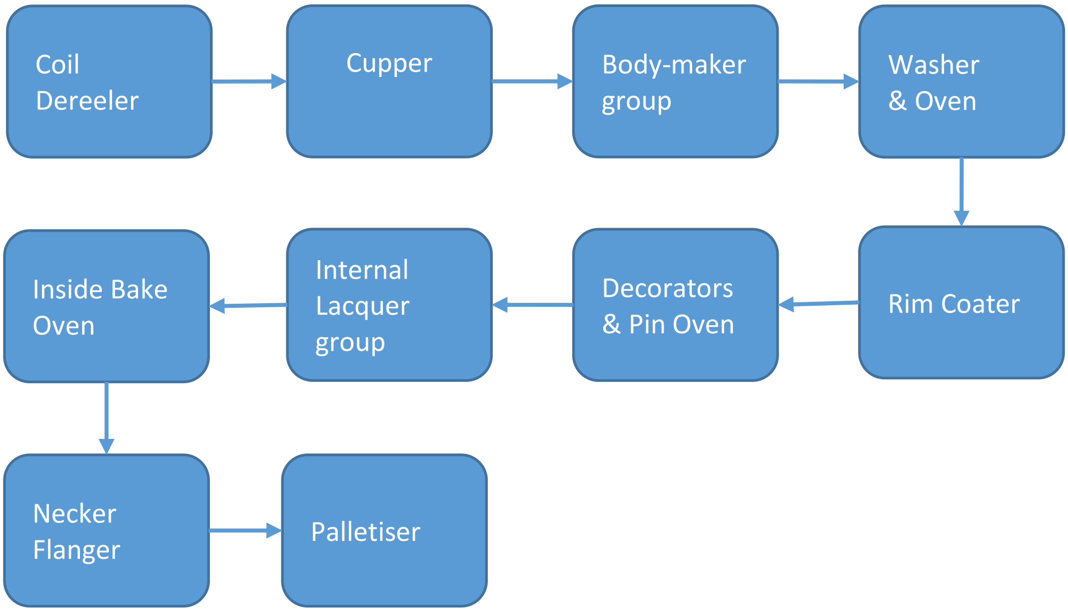


Figure 1 Beverage Can Manufacturing Processes

Each stage of manufacture is connected by high capacity conveyors. The conveyor systems contain additional buffers, BiDis which allow bidirectional flow, but these are mainly used as an escape system to allow ovens and washers to empty if the production line becomes blocked in front of them. The conveyors are controlled by a sophisticated PLC based "Line control" system which allows for machine speeds to be modulated depending on the build-up of cans on the conveyors (and buffers). The process flow for beverage cans and beverage ends is shown in Figure 1 and Figure 2 respectively.

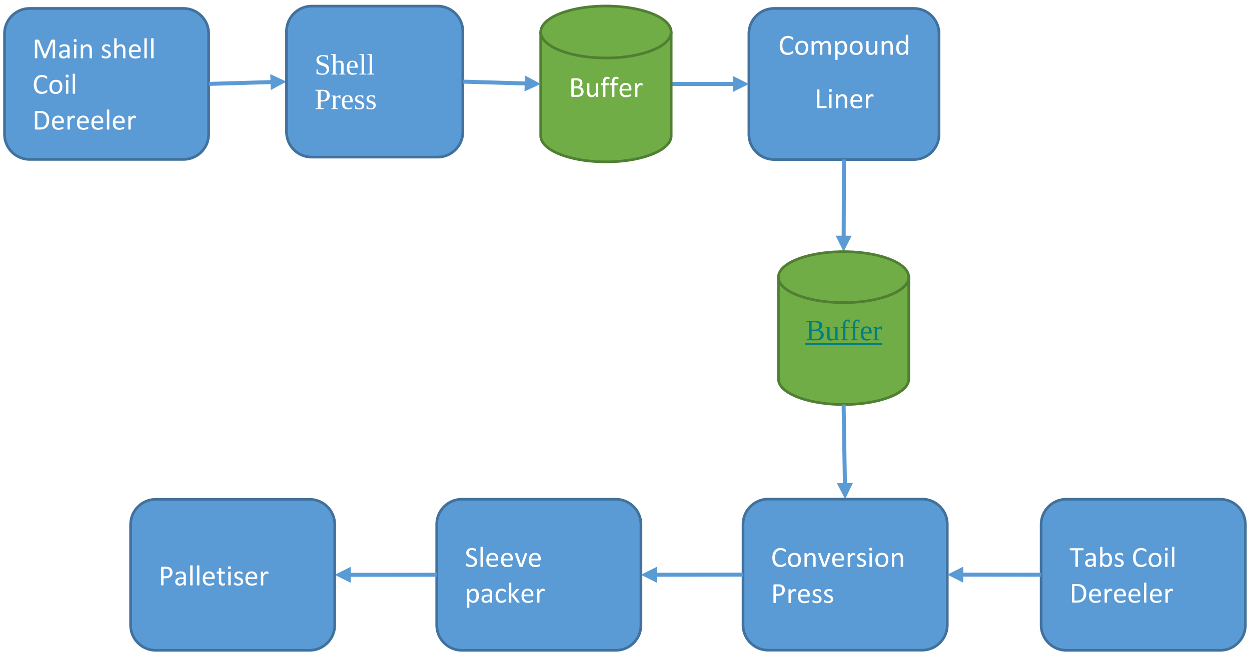


Figure 2 Beverage End Manufacturing Processes

1.2.2 Food

The food can business is smaller than the beverage business. The requirement for their customers to cook their product inside the can means that there has been much less scope for down-gauging. The production equipment is generally much older and the factories are much less well connected, digitally speaking, than the beverage factories. Although it is feasible to implement the machine monitoring software, Smartline, the high capital investment required to bring the PLCs and network infrastructure to the required level have presented a barrier to its implementation.

The process of manufacture of 3-piece food cans in that printing is performed on flat and then stored for later forming, see Figure 3.

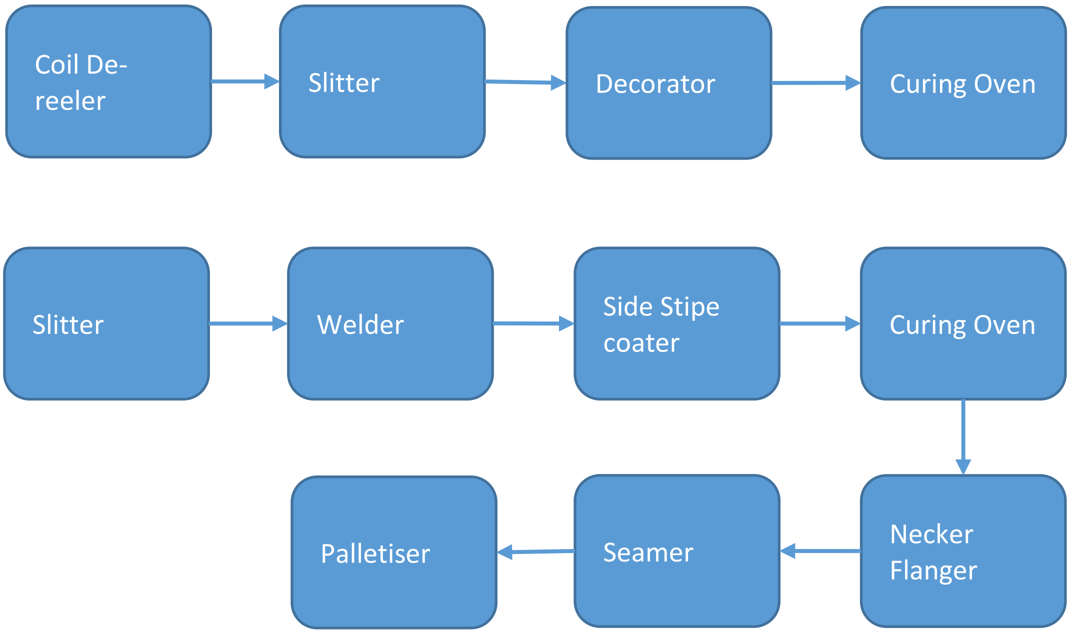


Figure 3 Food Can Manufacturing Process

1.2.3 Speciality Packaging

The speciality packaging business is similar to the food business in that the age of the machines has prevented them from implementing Crown's Smartline system for monitoring the production lines. The process is similar to food can manufacture in that the printing process is separate from the main process of tin manufacture, see Figure 4.

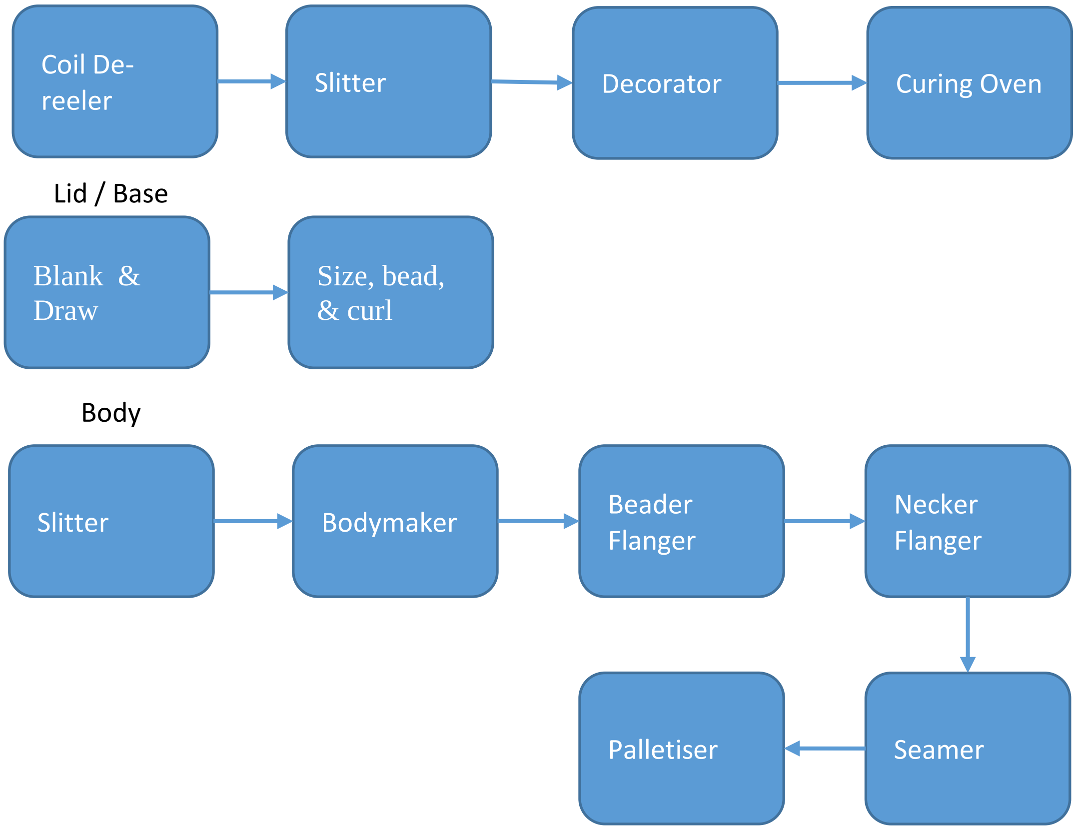


Figure 4 Speciality Packaging Manufacturing Process

1.3 Research Problem

Crown operates within a highly competitive market supplying beverage, food, aerosol, and speciality packaging with 143 manufacturing plants globally. They have noted that recent advances in technology might enable them to increase output of existing lines. An increase in production efficiency of 3% would provide the extra output equivalent to 3 or 4 factories.

1.4 Research Aim

The aim of the research is to show whether or not a digital twin based upon simulation, supporting more intelligent decision making, would allow a productivity increase thus achieving Crown's aim of unlocking the "Hidden Factory", from existing production lines

1.5 Research Questions

The research questions that arose were:

- How can simulation support Industry 4.0 and smart factories?
- How has simulation and Industry 4.0 techniques been implemented on high speed mass production systems?
- Can simulation be implemented into a semi-autonomous factory control system taking data from these systems to provide information to decision makers and ultimately achieving a fully autonomous control?
- How could a Discrete Event simulation based digital twin provide meaningful information?
- How can Industry 4 techniques provide benefits to Crown's manufacturing systems?
- What are the main barriers preventing Crown benefitting from Industry 4.0?

1.6 Research Objectives

1. Review of state of the art case examples pertaining to digital twin for high speed, almost continuous, manufacturing systems, particularly within the food and drinks and packaging sector.
2. To develop a baseline simulation model to investigate opportunities for theoretical improvement in the production systems, in order to create a quick win and obtain company-wide buy in and as a manual platform to develop an understanding of the requirements for a Digital Twin
3. Appraise the production line control systems for Crown's production facilities that the baseline simulation model is reflecting.
4. Establish whether the information is available from the control systems to supply input data to a digital twin simulation model.
5. Identify what other Industry 4.0 techniques and technologies would help Crown to reduce waste and increase throughput.

2 Methodology

2.1 Introduction

This section outlines how I designed and implemented my empirical research in order to achieve the objectives and thus answer the research questions.

This item has been removed due to third party copyright. The unabridged version of the thesis can be viewed at the Lanchester library, Coventry University

Figure 5 the research 'onion' (Saunders 2016)

2.2 Philosophy

I have chosen a pragmatic approach to the research. The DISCRETE EVENT SIMULATION experimentation will have quantitative elements and the interview will be qualitative. Although Saunders (Saunders 2016) argue that a more structured approach is preferred, there are number of reasons that I chose interview:

- Small number of line experts involved.

- The line experts are work colleagues and therefore anonymity is not an issue
- The subject requires insight and understanding with open questions and extended responses.

(Gillham 2000), show that in these circumstances, there is a benefit to the interview as a means of research. In particular the open nature of the interview allows for detailed answers and a dialog to ensure proper understanding of the interviewee.

2.3 Approach

From (Saunders 2016) the interview process is an inductive approach, where I hoped to gain a better understanding of the line controls system and then create a testable theory as a result.

The experimentation part of the research will be a deductive reasoning. I hope to obtain enough information from the line specialists to create a hypothesis, testable by Discrete Event Simulation experimentation.

2.4 Methodological Choice

As I will be collecting experimental data from Discrete Event Simulation, together with survey and literature review, it would seem that the methodological choice is “mixed method” (Saunders 2016). These methods will be triangulated within the Discussion section.

2.5 Strategy

The question of how a digital twin can provide meaningful information will be a combination of experimentation and literature review focused on Industry 4.0 use in high speed manufacturing and in the food processing industry.

The question of what has prevented Crown from making more Industry 4.0 initiatives will be answered by a combination of literature review and interview with line experts.

I hope to demonstrate that a digital twin could be of use to Crown by a literature review of general principles and demonstrating that the use could be applied to high speed manufacturing.

The experimentation with simulation software is a deductive approach in order to verify the theory that the changes to line controls can make a statistically valid difference. The DISCRETE EVENT SIMULATION experiments will be performed in using the industry best practice. The conceptual model is used to verify understanding of the real system in a non-coded human readable form in a way that the system experts can critique and verify the logic. The baseline model validates that the model mimics the real system with enough confidence to perform experimentation, see Figure 6 Simulation Cycle.

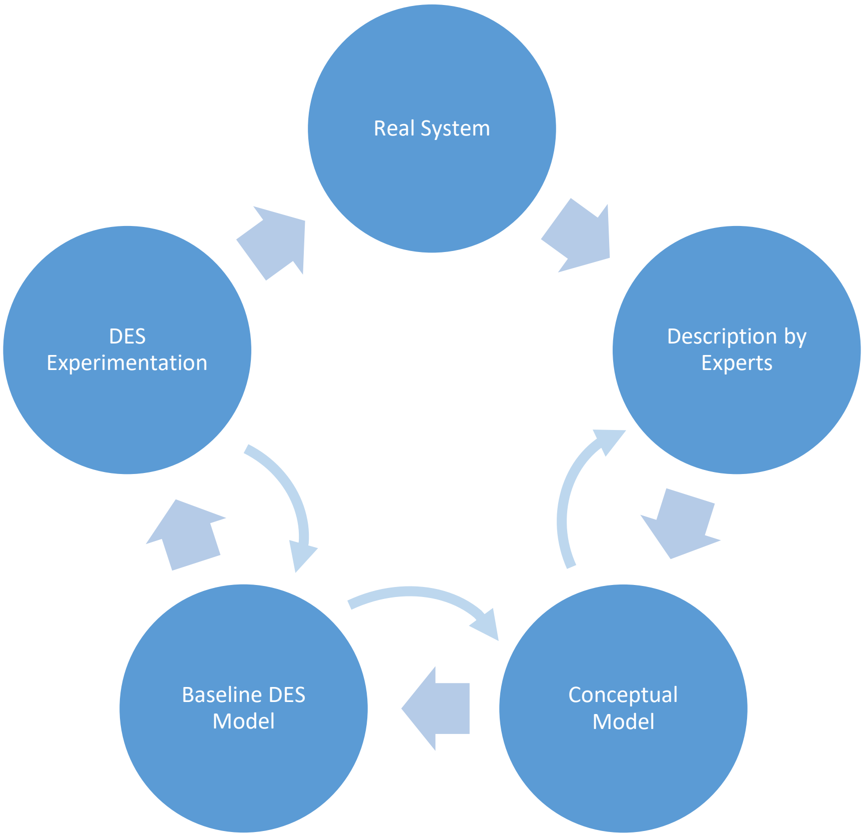


Figure 6 Simulation Cycle

2.6 Time Horizon

Cross Sectional. I will be taking sample data from several times as a method of checking the validity of the model, but no inference will be made as to the evolution of the system over time.

2.7 Conclusion

There will be a number of techniques utilised, cross referenced, in order to answer the research objectives, see Figure 7 Research Methodology Overview.

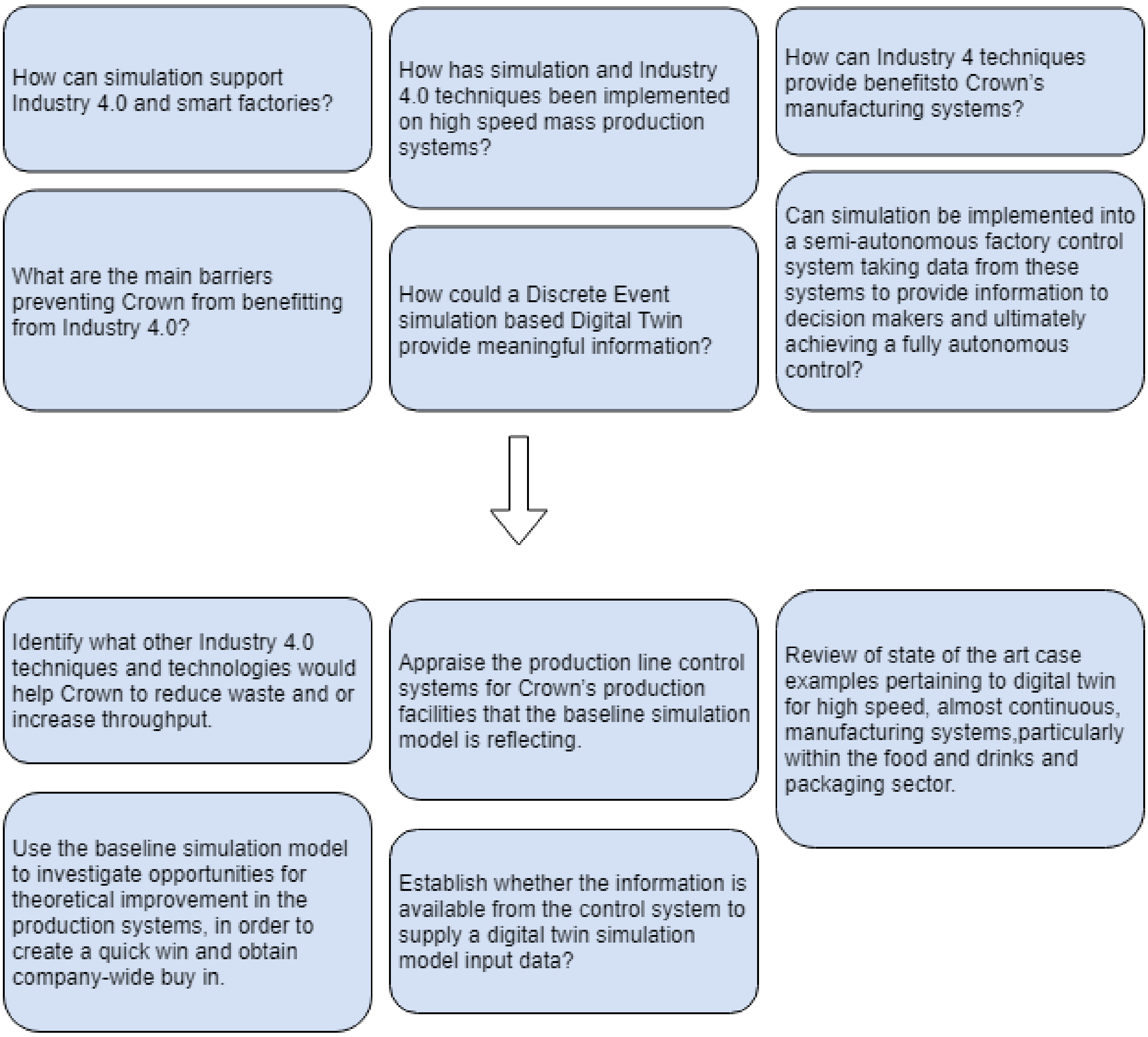


Figure 7 Research Methodology Overview

3 Literature Review

3.1 Simulation

Simulation is the experimentation with a simplified representation of an operating system as it progresses in through time in order to better understand and improve the system (Sharda and Bury 2011) , (Law 2015), (Zeigler and Oren 1979).

There are many advantages to experimenting using a simulated model as opposed to the physical operating system (Al-Bazi 2017) :

- Cost – although creating a model is not a cheap exercise, it is then much cheaper to explore many different options, which could be cumulatively expensive to investigate in the real system.
- Time – There may be a period of data collection, model construction, and validation as an overhead to simulation, but the time to explore many variants of the system will be much quicker than with the real system.
- Safety – simulation provides the tools to experiment where the environment generate dangerous scenarios. It is obviously better to explore and eliminate the potentially dangerous options in a virtual environment.

There are many Simulation techniques which include Monte Carlo, System Dynamics, Discrete Event Simulation, and Agent Based simulation. Each has its benefits and limitations.

3.1.1 Monte Carlo

Monte Carlo simulation is the method of linking a network of related events where the outcomes of individual events is a known in terms of a probability distribution for any factor that has inherent uncertainty. The model calculates the outcome of the primary events based on a random value from the distribution and feeds them down the network as inputs to calculate further dependant outcomes.

Originally developed to solve complex systems with coupled degrees of freedom and include randomness to solve deterministic problems. It was the method used to simulate the Uranium "decay" for Manhattan Project.

A single calculation thus creates a single possible outcome of the network. It then calculates the results over and over, each time using a different set of random values from the probability functions. Depending upon the number of uncertainties and the ranges specified for them, a Monte Carlo simulation could involve thousands or tens of thousands of replications before it is complete.

The output of a Monte Carlo simulation is not a single value but a distribution curve of the value, allowing the user to predict the most likely output of the real system and also the likelihood that the result will be valid. (Zio 2013)

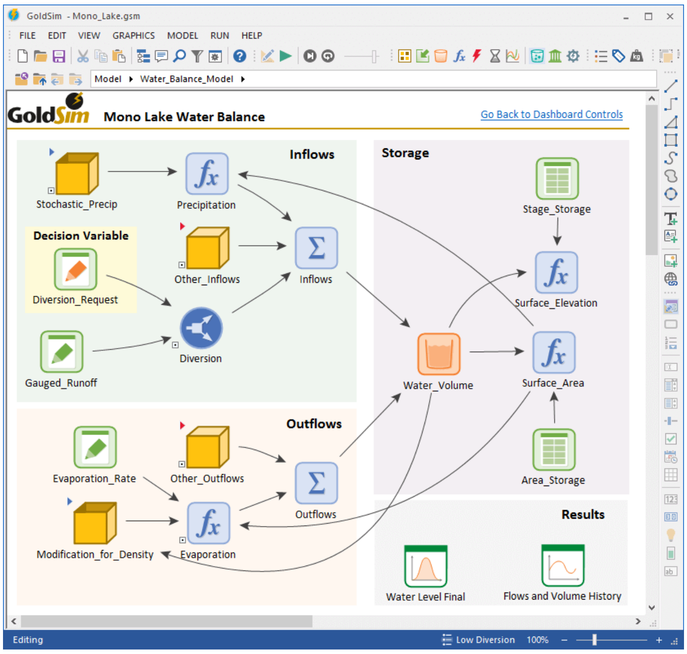


Figure 8 Goldsim Monte Carlo Software

There are several MS Excel spreadsheet add-in products to perform Monte Carlo analysis, for example @RISK, and products designed specifically, for example GoldSim.

The software is designed to create connections and feedback loops graphically. The feedback loops begin to look like system dynamics models.

Whilst there is some element of dynamics, where the simulation can model the real system over time, Monte Carlo analysis tend to be consider the steady state of the system.

3.1.2 System Dynamics

Unlike Monte Carlo, System Dynamics (SD) is a method for understanding the behaviour of complex systems over time. It was originally developed at MIT to model energy flow in engineering problems.

SD is based on the principles of kinematic and kinetic state variables (Fabien 2009) which are used to study the flow of energy through a system. The relationships being monitored have some kind of differential or integral relationship, for example velocity and acceleration. Figure 9 shows the linked relationship between velocity and acceleration. The constant acceleration produces a linearly increasing velocity, shown as the graph “Calc Velocity”, which is lined to the velocity valve. The change in velocity produces a squared increase in displacement over time, shown as the graph "position".

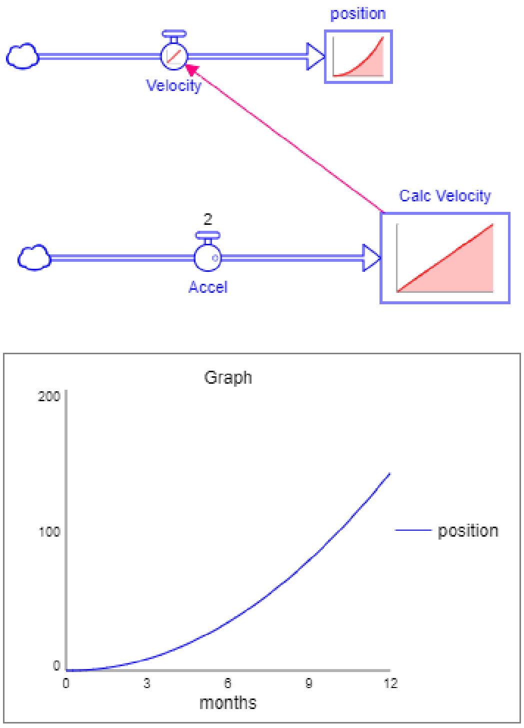


Figure 9 Simple System Dynamics Model (Stella)

Using SD it is easier to conceptualise and show the additional factors such as drag related to velocity and reduction in mass related to the duration of the force (as in a rocket burning its fuel) to the initial simple model, see Figure 10.

In order to break the circular reference between linked variables, for example drag and speed, SD codes employ a delay function to calculate one of them. The delay function has the effect of slowing the response of this first order calculation, but so long as the time delay is small compared to the simulated time and the responses being studied, then this approximation has a minimal effect upon the overall system.

In Figure 10 Enhanced System Dynamics Model (Stella), the drag is calculated as a function of velocity and the velocity is a function of acceleration and the acceleration is a function of drag. The drag is calculated using a delay function to remove the circular reference.

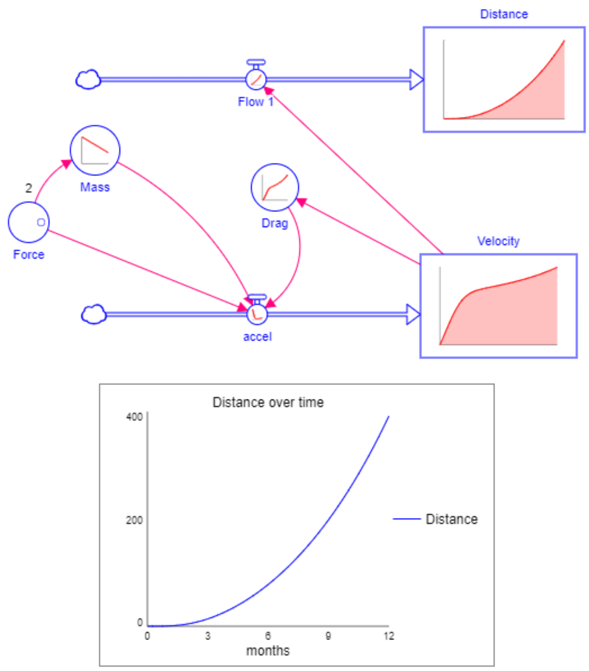


Figure 10 Enhanced System Dynamics Model (Stella)

The models are built in terms of feedback loops and time delays which affect the dynamics of the entire system. As well as the many applications for physics, the paradigm can also be applied to more abstract systems such as population growth, supply chains, and production lines where the system focusses on stock levels and rates of change of stock (Killingsworth 2011). System Dynamics is a useful tool to investigate the effect of long supply chains as orders cascade up and down the supply chain causing large oscillations in inventory levels. System Dynamics is used to test the Hypothesis of sharing inventory levels across the whole supply chain, rather than simply relying on immediate supplier-customer. To dampen the oscillations (Croson and Donohue 2006). (Georgiadis and Michaloudis 2011), similarly uses System Dynamics to improve the supply chain dynamics.

System Dynamics can be used to model the effects of machine reliability and manpower absenteeism if the causal loops can be identified. In the paper (Gupta, Narayanamurthy and Acharya 2018) SD is used to model the waste produced in tyre manufacture and to identify the opportunities to implement lean manufacturing. They demonstrated that the cost of increasing the skill level of the workforce, was much less than the cost of rework and waste produced.

There are papers (Feng and Fan 2013; Georgiadis and Michaloudis 2011) that show production line analysis is possible using System Dynamics, but both papers pay attention to the systems stock levels rather than queueing and starvation effects.

The technique is useful for capturing the dynamic effects of a system over a time period, but will not be able to catch the queues and gaps in the system that lead to less than expected output of a system (Brailsford et al. 2014).

Because of the overview nature of SD models, a great deal of care must be taken to validate the integral relationships within the model before experimentation.

3.1.3 Discrete Event Simulation

3.1.3.1 History

Discrete Event Simulation is the technique of modelling an operating system in terms of its capacities, processing times and buffer capacities. Rather than representing stock levels as a single variable, Discrete Event Simulation represents individual items of stock as temporary entities. Permanent entities are used to represent the machines in the system. The technique uses random number distributions to calculate various times the tempory entities spend at each machine and in this way is able to reproduces the queues and gaps in the real system. The system totals key performance indicators as the temporary entities exit the simulation and in order for the key performance indicators to be meaningful, they are averaged over a long period of virtual time or across many replications.

It is based on queue theory originally developed in 1909 by Agner Krarup Erlang (Medhi 2003) which described a simple operating system by breaking it down into basic characteristics of inter arrivals of customers; pattern and the number of servers. Queue theory serves to calculate the performance of the system, calculating throughput and average waiting times for non-deterministic arrivals and service patterns. Queuing theory has limitations in terms of the complexity of the system being modelled. Discrete Event Simulation overcomes this shortcoming by modelling the operating system directly, and averaging the individual results of objects passing through the system.

The technique was one of the first to be computerised in the 1950s and Robinson (Robinson 2011) divided the development into four distinct eras for Discrete Event Simulation, to use his terminology, these are: “Pioneers” in the late 1950s and 1960s, “Innovation” in the 1970s, “Revolution” in the 1980s, and “Evolution” from the 1990s to 2000s.

The Pioneer era, which predates modern computing, Robinson relates how Discrete Event Simulation models were often written in machine code for speed and memory concerns, and were pre-processed using punched cards. The output in the form of a printed numerical summary could only be analysed by specialist statisticians. Iterations were both expensive and time consuming exercise, both in computing and manpower. In the 1960s as computer speed steadily increased, Discrete Event Simulation became a well-established process, general purpose Discrete Event Simulation software became available (e.g. GPSS, SIMSCRIPT and SIMULA). In the 1960s research by Tocher (Tocher 1967) lead to the specification of the three-phase approach to remove system deadlocks, which remains the basis of many modern simulation packages.

The Innovation era, as computer processing speed jumped significantly due to commercially available microprocessors, and Discrete Event Simulation packages were developed (e.g. SLAM and GPSS-H) to reduce the model creation time, but the process was still considered expensive by modern standards, both in cost of computers and manpower.

In the Revolution era, industry standardised to the IBM personal computer and MS-DOS and eventually MS Windows operating systems. The standardisation provided software designers with a large and ever expanding market, which allowed them to develop sophisticated general purpose Discrete Event Simulation software. The progress moved towards graphical systems, firstly visually interactive software (VIS), and eventually visual interactive modelling systems (VIMS).

In Robinson's Evolution era, computing speed seemed to increase exponentially even as real costs fell, which allowed the Discrete Event Simulation VIMS to become more refined and reliable and capable of analysing more elaborate systems. New software companies entered the Discrete Event Simulation market with the cost of basic software in the region of £1000.

The point of Robinson's article was that as Discrete Event Simulation software has become cheaper and much easier to use, there is a danger that without the relevant training, the output could be misleading. Robinson cited evidence that Discrete Event Simulation is increasingly practiced by persons without the experience or training to employ the scientific rigor required. A survey of Discrete Event Simulation users showed that a third of the respondents were using fixed length of warm up and half of respondents did no validity checks on the replications of runs (Hollocks 2001).

It could be argued that Discrete Event Simulation is about to enter a fifth era, with concepts of "symbiotic simulation" (H. Aydt et al. 2009), "digital twin" (Greaves 2014; Robinson 2011), and "Cyber-physical production Systems" (Thiede, Juraschek and Herrmann 2016) in which Discrete Event Simulation software is linked in real time to the physical production system.

3.1.3.2 World View

In Discrete Event Simulation rather than represent stock levels, as in the System Dynamic approach, items of stock are represented by individual temporary entities flowing through the system. Permanent entities represent the servers in a system and tend to be locations where resources are consumed by the tempory entities and where the "value added" processing is calculated. (Brailsford et al. 2014) gives an excellent overview of the Discrete Event Simulation technique. Whereas System Dynamics models the system from an overviewing perspective, Discrete Event Simulation models from the bottom up modelling objects arriving into the operating system and totalling the time spent in transit; waiting in a queue; and being served. The system aims to replicate directly the queues and gaps in the actual system caused by irregular arrival and processing times (Brailsford et al. 2014).

For example, considering a simulated queue of customers waiting in line to be served in a supermarket, the objects of interest would be the checkouts, classed permanent entities as they exist for whole simulation and the customers arriving at the checkout, classed as temporary entities as they appear from the supermarket and disappear to the exit during the simulation. The states for the checkout would be busy serving a customer, idle waiting for the next customer, or on a break where the checkout becomes unmanned. The states for the customer would be waiting in the queue or being served. The events would be customer arriving into the queue, customer moving from the queue to the checkout and then leaving the store. The events for the checkout would be changing from idle to busy. This detailed level of modelling allows results to be generated from multiple perspectives, for example in the supermarket model results can focus on either the checkouts to determine how much time is wasted waiting for customers to arrive or it can focus on the customers, estimating how much time they spend queuing for service. The system introduces an element of chance by representing arrival events and process times by a mathematical distribution driven by a pseudo random number. In the book (Allen 2011b) the practical aspects of random numbers and distributions are discussed. Running a simulation for a short period of simulated time, does not produce results with any practical value. If the simulated system does not have an end condition, then it can be run until the results, not the simulation, reach a steady state (Whitt 1991). If the system does have a terminating condition, for example the supermarket closing, which is less than required for the results to reach a steady state, then multiple replications with different random number seeds can be used and averaged.

Discrete Event simulation, as the name implies, is driven by events to advance the simulation time clock rather than running a continuous system clock and dealing with events as they are met. This has the advantage of the system being able to jump large blocks of time where there is no activity. In order to remove the possibility of deadlocking, a three phase approach (Choi 2013; Pidd 2004) was proposed by Tocher in 1963. In it he proposed two type of events:

- “B” - Bound or booked, based upon a time event, for example the next entity to enter the system or server completing a process.

- "C" - conditional Events, for example an entity waiting for a resource to become available.

The three phases of the program are processed in order and then the system loops back to the top:

1. Phase A: Advance the simulation clock to the next B event.
2. Phase B: Process the B event
3. Phase C: Scan the C events and process where possible

The "B events" tend to be events which release resources and the "C" events tend to consume resources, so processing the B events first each cycle makes logical sense. The three phase approach is a fairly efficient programming algorithm with the main inefficiency being the time to scan of all the "C" events in order to determine if they are runnable.

In the supermarket example a "B" event would be the calculated arrival time of the next customer or a checkout finishing serving a customer. All the other customers in the queue are examined as the "C" events. The customer at the head of the queue moves up to the checkout and commencing service. The time to service is immediately calculated and inserted into the ordered list as a "B" event.

Whilst the three phase approach has been successful, it is one of the main barriers to implementing a Parallel Discrete Event Simulation code (Chen 2015; Fujimoto 1990).The problem of executing B events out of order creates causality errors where the value of a state variable is being modified by more than one event representing more than one time. One solution to this is to split the problem into geographic domains which only need to pass messages at certain locations (Hou et al. 2013). The overhead of passing messages between domains will undoubtedly mean that the performance will not scale directly with the number of CPUs but the nature of Discrete Event Simulation as a method, as opposed to the software, will normally require multiple replications using different pseudo random number seeds, therefore running the replications concurrently, will speed up the overall analysis time.

3.1.4 Agent Based Simulation

Discrete Event Simulation codes involve generic entities following rules programmed into the servers which change their status and cause the entities to queue or otherwise change state. The entities themselves do not make decisions. Agent based simulation is similar to Discrete Event Simulation, in that individual entities (or groups) are modelled. The main difference is that the entities communicate with each other and have goals to satisfy, rather than satisfying the goal of the system

Although the concept of agent based modelling has been around since the 1940s, SWARM was arguably the first serious computer implementation in the mid 1990s (Allen 2011a). It has been used be to simulate single celled animal populations (Csonto, Kadukova and Polak 2001) where the agent rules are very simple but lead to complex emergent behaviour. Swarm was also used to model supply chain networks with the aim of reducing inventory costs (Fu-Ren Lin, Gek and Shaw 1998) .

Agent based simulations give the tempory entities intelligence (Allen 2011a; Fu-Ren Lin, Gek and Shaw 1998), or at least an arbitrary choice mechanism which can lead them to adapt to changing circumstances. This situation is relevant for many problems where organism populations, including humans, are represented in the system, although there are papers where the intelligence being simulated is artificial. SimIShopF, an agent supported simulation tool, is used to simulate the complex manufacture system where the agents have a specific manufacturing schedule but have autonomy over which resources to claim (Rs Chen 2003; Ruiz et al. 2011).

This programming paradigm has been introduced into many Discrete Event Simulation software programs (Kehl ; Zankoul, Khoury and Awwad 2015; Zupick 2016) .

3.1.5 Simulation Software Selection

Production line problems are caused by the complex interaction of events, best captured with Discrete Event Simulation. The lines to be simulated are very linear (see Figure 1 Beverage Can Manufacturing Processes) and therefore fitted the Discrete Event Simulation paradigm, see Table 1 Advantages and Disadvantages of Alternate Simulation Methodologies

	Advantages	**Disadvantages**
Monte Carlo	Fast run time.	Individual interactions are not modelled.
System Dynamics	Fast run time.	Individual interactions are not modelled.
Discrete Event	Accurate interactions	Long Run time for complex models
Agent Based	Accurate Interactions. Ideal for entities with decision making capabilities.	Long run Time. Complex…

Table 1 Advantages and Disadvantages of Alternate Simulation Methodologies

Simio was chosen as it is based on object oriented programming paradigm advantages. It is able to run experiments and replications in parallel so that overall run speeds. Simio has Agent Based Simulation capabilities, see (Kehl), and therefore is a good fit for introducing extra complexity of human choice interactions, if necessary, see Table 2 Advantages and Disadvantages of Discrete Event Simulation Codes.

	Advantages	Disadvantages
Arena	Relatively difficult to model structure	Known to Coventry university
Simio	Agent based simulation capable. Object Oriented programming approach. Parallel execution of replications.	No direct experience
Simul8	Unknown	
Plant Simulation		Expensive
ProModel	Extensive user base, good support, maintenance… Expensive.	No direct experience
Open Source	Potentially fast	Language based, with long model development time

Table 2 Advantages and Disadvantages of Discrete Event Simulation Codes

3.1.6 Simulation Hardware Selection

The choice of Simio has allowed for running of experiments and replications in parallel. The experiments have to be model parameters, although this is simple to achieve and even alternate data-sets are available as model parameters. The software manages all the replication runs for each experiment and collates the results to export to Excel. In order to increase the model throughput, the software was installed on a workstation with a 10 CPU (20 hyper threaded) processor with 64 gigabyte of RAM which allowed 20 replications or experiments to be run concurrently. The computer is remotely accessed using HP Remote Graphics Software (RGS) which allows graphical access from Crown network, including via virtual private network (VPN). It gives some of the benefits of web based modelling, such as access to much greater processing speeds and a system that is always running, described in the paper (Byrne, Heavey and Byrne 2010), without the considerable cost of Simio cloud based processing.

3.2 Industry 4.0

There is a consensus that industry 4.0 will have a large positive impact on the global economy. According to Aitken the value is massive, "Boston Consulting Group anticipates that by 2020, the digitization of manufacturing processes, connected supply chains and new business models will add $1.3trillion to global economies" (Aitken 2017). With even conservative estimates, businesses will benefit from the greater insight to their business operations. Aitken accepts that many industries will have to make investments within their limited resources and that most companies will experience Industry 4.0 as evolution rather than revolution.

The internet is growing in every conceivable direction Figure 11. . Growth has been vertical, for example internet based shopping, which started as a niche in the form of EBAY and Amazon, are now employed by most retailers (Gilchrist 2016). Horizontal growth is where the internet has begun to affect different aspects such as the industrial internet. It could be argued that fitness monitoring is a new aspect, but it is firmly driven by supply and consumer demand for such devices. Industry 4.0, sometimes referred to as “Industrial Internet of Things” (IIoT) refers to the introduction of internet derived technologies in to manufacturing facilities.

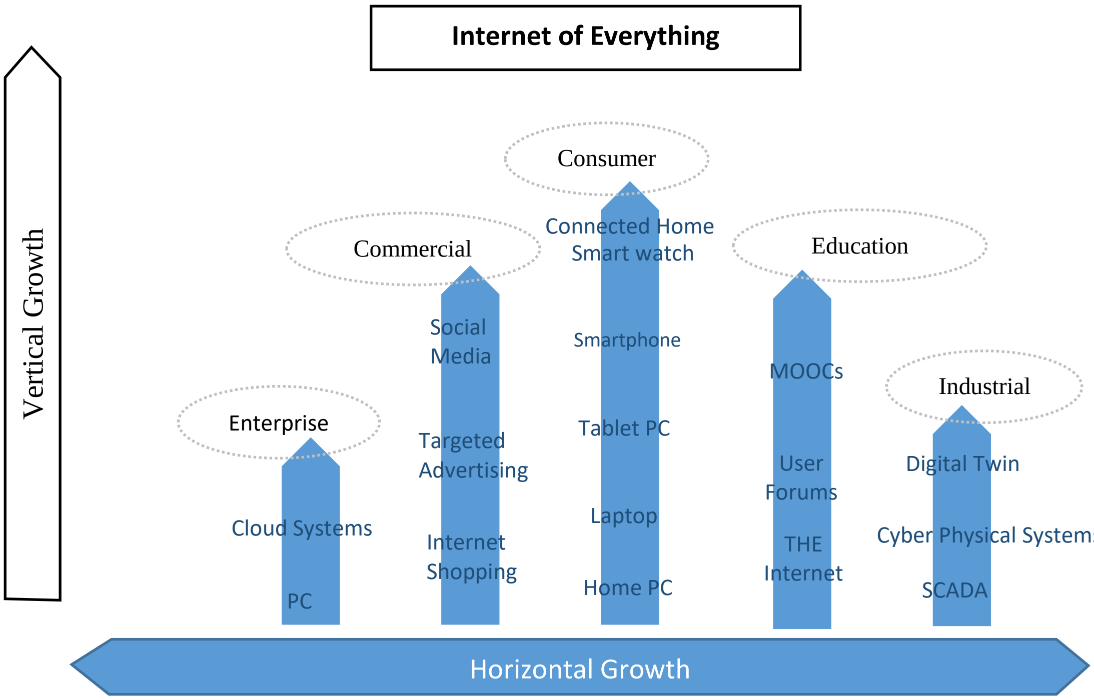


Figure 11 Internet of Everything

There is a consensus as to the definition of Industry 4.0 in terms of levels of automation. It is generally accepted that Industry 1.0 is defined as mechanisation and powered systems; Industry 2.0 is mass production with standardisation of components; Industry 3.0 automation and Industry 4.0 is the implementation of analytical systems. Figure 12 Industry 1.0 to 4.0 shows the relative levels of industrialisation.

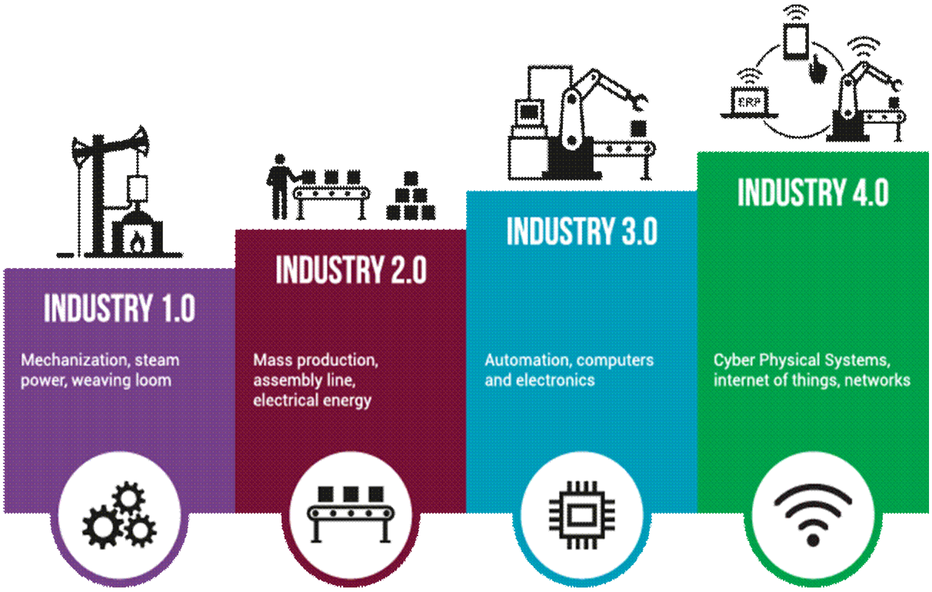


Figure 12 Industry 1.0 to 4.0

The architecture of Industry 3.0 is fundamentally different to Industry 4.0. Whereas in industry 3.0, machines may communicate with each other, the connections are enabled by specialist programmers making bespoke software to connect two or more different devices. Typically in this configuration, the connections are solely real time and for the purpose of cooperation between the machines.

Industry 4.0, the paradigm is very different, the machines are connected through middleware, and the connections between machines occurs at the level above this, which allows for data capture and storage for later analysis. Whether this occurs at the internet level or intranet level or at the edge level, the protocols are the same.

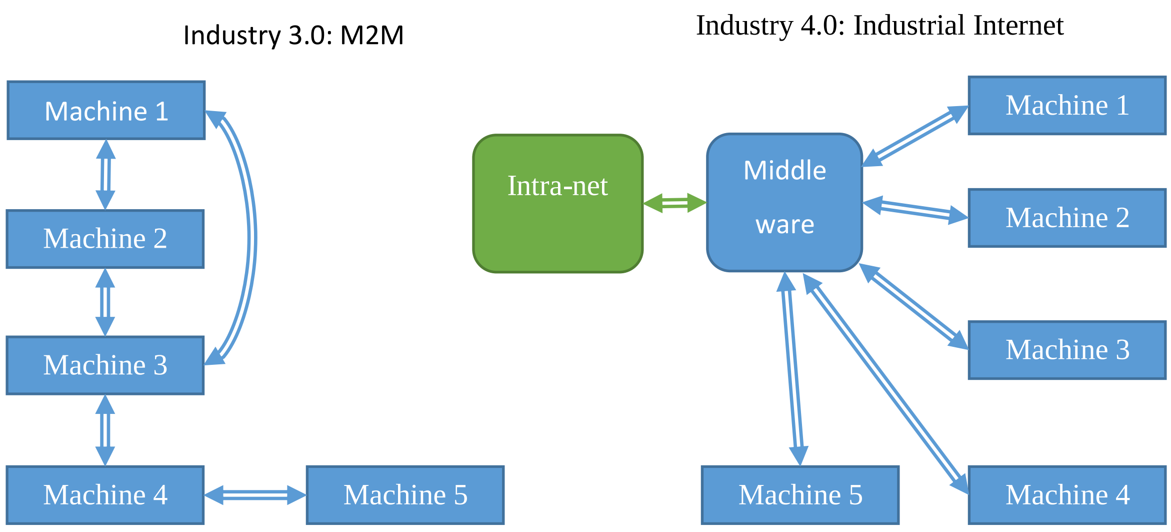


Figure 13 Industry 3.0 vs Industry 4.0

The middleware layer should be as open a standard as possible so as to eliminate the lock in costs of a single supplier (Liu and Jiang 2016; Raggett 2015)

A paper discusses the opportunity of embedded sensor technology aimed at monitoring and improving shelf life of food packaging (Schaefer and Cheung 2018). This is interesting, but with canned food shelf life measured in years, there seems little market value to an even longer lasting can.

3.2.1 Edge Computing

Edge computing is defined as technologies that allow computation to be performed near the data source (W. Shi and S. Dustdar 2016). There has been much discussion with regard to where the intelligent control should be placed and there are conflicting views (B. P. Rimal, D. Pham Van and M. Maier 2017). If the intelligence is placed on the cloud, or intranet server, then only one complex system is required to be maintained; if intelligence is placed at the edge then there is low latency of response.

If the intelligence is placed at the edge, then there are many advantages:

- System is not as reliant upon stable network connection as it can have fall back procedures in the event of communication loss.
- Data can be acquired at higher rates, without contention or bandwidth limitation
- Data can be compressed and aggregated before sending to the cloud, this can massively reduce the network traffic see Figure 14.

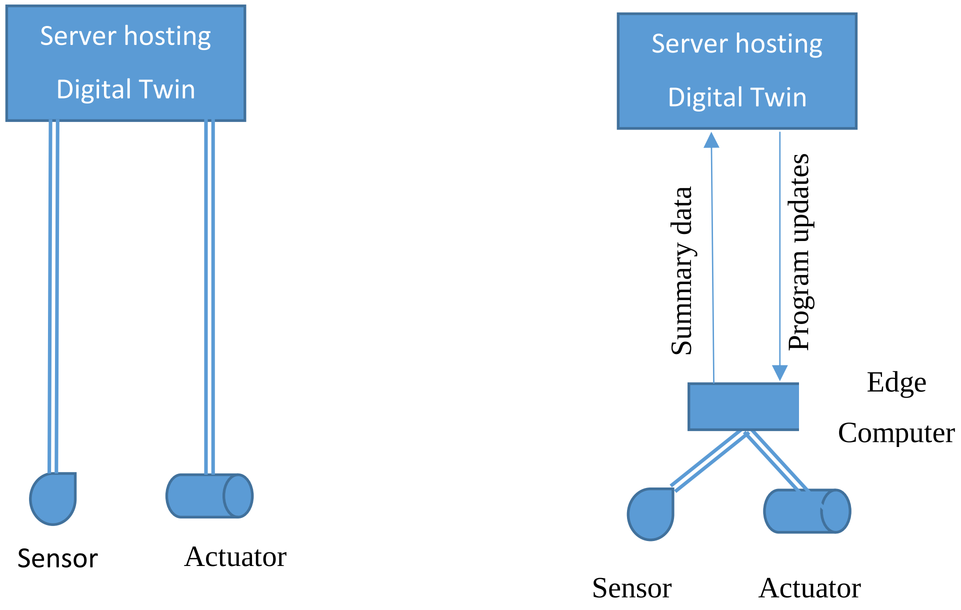


Figure 14 Cloud versus Edge Control

There is also two more useful features of distributed intelligence, often utilised in ultra-low power requirement scenarios (Baran 1964):

- In order to conserve energy, the edge devices schedule the data transmission
- Where there is no network to latch on to, the devices could form their own network acting as nodes to ensure communication with the furthest device see Figure 15.

Figure 15 Centralised, Decentralised and Distributed Networks (Baran 1964)

Cloud computing is an efficient way to process huge volumes of data due to the massive processing power of modern server computers. However some systems require real time feed-back to devices which produce vast quantities of data but data transmission speed and network latency do not allow for cloud computing. An extreme example would be an autonomous vehicle with a camera vision system: the decision time is in the order of milliseconds and the raw data from the vision system is in the order of megabits per second. In order to utilise a vision system, an autonomous vehicle must process the data from the vision system locally. As an industrial development example, it was demonstrated that a useful feedback system implemented at the edge (Trinks and Felden 2018), removed the latency problems of control from the cloud.

That is not to say that Edge Computing is a technology which replaces cloud computing, it is best designed to be used in conjunction with cloud computing by delivering a rich user experience and collating the data into manageable chunks for further processing in the cloud system. An effective example of this is mobile device satellite navigation system (Nightingale 2017): They work in in conjunction with the cloud by processing the data locally on the device to generate the instructions and graphical maps to the user. The cloud is the source of the local map data, as they are designed to work anywhere in the world, and provides higher functions such as traffic awareness. By working together, the user is given the illusion of seamless navigation data for the whole planet on a small computing device.

Industrial applications for the above point are contained in papers using the terminology FOG computing, where the cloud is close to the ground (Bonomi et al. 2012; Gómez et al. 2015) .

An well as adding to the user experience, edge computing being can be used as a low cost alternative to adding wired network infrastructure by performing the task of networking as well as data collection as in this example (Gómez et al. 2015), although since the paper was written, there are a plethora of industrialised low power edge computers have become available, included an industrialised version of the Raspberry Pi available commercially. Google have recently developed "Android Things" which is aimed at giving artificial intelligence to low cost computers including the Raspberry Pi. It was used to retrofit low cost remote sensors to a grain warehouse (Chibuye and Jackson 2017) . It serves as a good example with a lot of similarities to Crown's older factories:

- The hardware is cheap with good input/output to connect to analogue sensors.
- It is license free so will scale up without cost.
- Wifi connectivity meant little cost to site the device in the correct place.
- It was designed to sit on an unreliable Wi-Fi network with relatively low bandwidth to the main server.

The key benefits of edge computing in a manufacturing context are reiterated in the short article (Mhetre 2018): Faster response time due to zero latency; Reliable control on unreliable networks due to the autonomous control systems; Security and compliance due to edge computing's technology; Cost effective due to no requirement for wired networks; and interoperability as they can act as communication liaison between legacy and modern devices.

3.2.2 Big Data

At Crown, manufacturing efficiency is based upon constant machine availability. The longer time between failure and shorter repair times, will lead to more profit. Scrappage on the line is also an issue, although the low cost of WIP it is of lower concern.

Increases in productivity are achieved by solving visible issues, for example breakdowns and spoilage. These issues tend to be internal issues, with causes and solutions being continuous improvement projects and applying best practices. In order to reach beyond the visible problems, Lee (Lee et al. 2013) asserts that manufacturers can adopt two methodologies, see Figure 16 Productivity Opportunity Space (Lee et al. 2013):

1. Work more openly with suppliers to find possible external causes for high spoilage.
2. Smart sensors recording performance linked across the production line.

An example of the second point would be to add extra sensors to a suspect process, and processes upstream of that process to see if there are any cases that could have unknown causes. It could be argued that this is an example of Continuous Improvement, but the added communication layer afforded by Industry 4.0, to send the data to the cloud, gives the opportunity to delve deeper and wider into the data by clustering machines.

This item has been removed due to third party copyright. The unabridged version of the thesis can be viewed at the Lanchester library, Coventry University

Figure 16 Productivity Opportunity Space (Lee et al. 2013)

The manufacturing sector has been regarded as slow to adopt internet technologies compared with other industries which is due to need to deal expensive to replace legacy equipment (Babiceanu and Seker 2016). In recent years, advances in sensor technology, driving cost down and connectivity up, mean that retrofitting is an established technique (Guerreiro et al. 2017).

The technique of clustering, where identical machines both at a line, factory and global level are grouped and compared allows for much more data to be considered (Lapira 2012). Additionally the technique of fuzzy clustering (Frieß et al. 2018) allows for non-identical machines, which are performing the same function and behave in a similar enough way can be compared.

3.2.3 Smart Conveyor

The need and application of smart conveying system have been around for much longer than Industry 4.0. The article explains how smart conveying systems (Murray 1997):

- Reduce maintenance costs by employing build back sensors and intelligently slowing heavy goods to remove collision stopping events described as “maintenance headache”.
- Reduce energy use by 25% to 75% by employing direct drive rollers and directing power only to rollers requiring power.

In terms of the factory of the future “Conveyors have become an essential component of the manufacturing process and not an afterthought that provides transport only for products” according to (Weber 2018). This is particularly true for Crown with so much work in progress on the conveying system. The article notes a move from mechanical pneumatic conveyors with little or no feedback towards electrical actuators which are claimed to be safer, smarter, and quieter with predictive maintenance built in.

“Smart factories are developing conveyance systems that can make smart decisions that track, sort, merge and accumulate product” (Weber 2018) .

A case study of retrofitting industry 4.0 technologies to bottling plant conveyors is outlined in the article (Kahiomba and Wang 2018). The paper describes fitting vibration sensors and to provide a closed loop monitoring software at the edge and cloud based predictive maintenance of the conveyor drive motors.

3.2.4 Computer Vision

One of the areas where Crown has highlighted a deficiency is with counting entities (cans or ends) on mass conveyors. This could be achieved with the implementation of a computer vision based counting system.

Computer vision has been the subject of much research since the 1950s (Nilsson 2010). OpenCV was first released as an alpha version in 2000(Brahmbhatt 2013) (Suarez 2014).

The OpenCV system includes shape detection, and specifically circular object detection using the Circular Hough Transformation (CHT). With additional processing, the method can be used to detect overlapping objects (Jianjun et al. 2016). Whilst it is unlikely that Crown would wish to count overlapping entities on mass conveyors, it forms a robust system to count entities that are touching and could be confused as a single entity.

OpenCV has been used successfully to count multiple moving entities (Seenouvong et al. 2016) (Abbas et al. 2017) (Suryatali and Dharmadhikari 2015). In order to count moving objects, the system needs to operate at close to frame capture rates, ie ~30 frames per second. The amount of processing per frame reduces the processing rate (Kun and Vamossy 2009) and therefore the accuracy of the count for an individual frame is compromised in favour of attaining a good frame rate. There are mitigation strategies, for example choosing a monochrome camera with better optics rather than high resolution as often these are the first transformations performed in software.

Computer vision is not limited to optical images and could be augmented with the use of a depth sensor, for example in the form of Microsoft Kinect system. As the depth is represented as a 2D image of depth, it can be processed as if it was an ordinary image. There have been several papers using this technique (Coskun et al. 2015) (Ching-Tang Hsieh et al. 2012) demonstrate counting people passing below a ceiling mounted Kinect Sensor and processing the depth video using openCV. The paper demonstrates anecdotally that the system is effective at distinguishing individuals in a crowd, but lacks data to prove the systems accuracy. Studies using similar hardware for crowd counting (Li et al. 2016) (Chen, Henrickson and Wang 2016) demonstrate an accuracy of 93% to 97%.

The basic principle for motion detection counting

1. Background training
2. Masking – areas not part of the counting region are deleted from the image.
3. Motion detection – subtracting the background image from the current image
4. Object detection and labelling – This ensures that the objects are not counted more than once. Where the objects being counted do not have any recognisable shape, this consists of creating a binary image and counting the regions of connected pixels above a threshold area known as binary large object (BLOB) counting. BLOB counting can miscount where two objects are close together in the image and do not form distinct BLOBs or when changes in illumination create extra BLOBs (Kun and Vamossy 2009).
5. Tracking and counting the objects across an exit region.

The problem of poor object detection can easily be mitigated for Crown by utilising the CHT method to count circular objects, rather than BLOBs

3.2.5 Digital Twin

The concept of a twin in order to perform offline experiment is not new. The Apollo space missions had a complete command module, complete with all supplies available to the actual crew on hand to explore options should the need arise and famously utilised on the Apollo 13 mission (Reinders 2018).

The term Digital Twin was originally coined by Grieves (Greaves 2014) . His definition “virtual models of physical objects are created in a digital way to simulate their behaviours in real-world environments” can relate to two types of Digital Twin. The first kind is where the Digital Twin is used for the design of the product and includes finite element analysis (FEA), product ergonomics, and production line layout. The second is where the live data from an operational production line is linked to a digital representation in order to drive day to day production decisions.

Aitken extends this to three levels: Asset level, Operational Process level, and Enterprise Business level (Aitken 2018). The asset level is split by Qi to give to give 4 levels of Digital Twin (Qi and Tao 2018) :

1. Product Design
2. Manufacturing Design
3. Usage Monitoring
4. Smart Maintenance

The distinction between 1 and 2 is useful as it allows for other simulation techniques for example finite element analysis (FEA) and kinematics in the Digital Twin worldview.

The first two represent the state of the art for good design with iterations being made in a virtual environment rather than physical prototypes and do not really represent a symbiotic relationship between the physical and virtual world. They may be revisited for future iterations but these versions of the twin are largely redundant once the product is in manufacture.

An example of Product Design Digital Twin was presented at the Applied Visualisation Forum conference. The lecturer enthused about the use of a Digital Twin to create a “production prototype” (Leeming). The use of multiple technologies coming together in a digital design that is industry 4 ready whilst impressive, is not a Digital Twin in the true sense of the meaning. The crux of the design process was a FEA kinematic model, which will never be used again in the life of the machine – it was simply a design tool. The reuse of CAD data to produces an animation of the machine in operation, is an important marketing tool, but did not include the FEA, and I am sceptical that the crucial part of the process, the wrapping process, was not tested by a physical prototype in advance of the production design. I am certain that the machine is fully instrumented and will indeed become part of a type 3 or 4 Digital Twin, by good design.

A paper (Nikolakis et al. 2018) describes an example of the Digital Twin implemented for the purpose of design. Its main focus is to improve pick and place warehouse operations by using sensors placed on human subjects to capture exact motions. They used actors perform the pick and place operation, rather than capturing the data from actual warehouse staff, so as to bypass the ethical concerns, but serves as an example as to how data could be collected.

The type 2 Digital Twin is now an accepted part of the project manager toolkit for mega projects such as Heathrow T5, Crossrail and HS2 (Whyte 2019). The Digital Twin in these cases forms an important visualisation for promoting the project to a wide audience to gain funding, see Figure 17 Liverpool Street Station Visualisation (Crossrail).

This item has been removed due to third party copyright. The unabridged version of the thesis can be viewed at the Lanchester library, Coventry University

Figure 17 Liverpool Street Station Visualisation (Crossrail)

A more detailed version, although less compelling visually, is used to plan the implementation and ensure that the large pieces of equipment can physically fit through the tight spaces, see Figure 18 Crossrail's Liverpool Street Station Planning (Crossrail).

This item has been removed due to third party copyright. The unabridged version of the thesis can be viewed at the Lanchester library, Coventry University

Figure 18 Crossrail's Liverpool Street Station Planning (Crossrail)

The information used in the construction of these projects will be used to create an operational Digital Twin "The replication of the physical by digital information is critical as the virtual railway will be used to manage and maintain the physical for its projected life of at least 120 years"(Anon.2016; Peplow 2016; Taylor 2018).

The third and fourth type of Digital Twin actually utilise the data from the physical system as an input to the Digital Twin. Lee also alludes to the view that the Digital Twin must also be linked in real time to the physical system to be considered a true Digital Twin (Lee, Bagheri and Kao 2015), his diagram shows the Digital Twin as the next logical step following connection with smart sensors.

The methodology of an autonomous Digital Twin is demonstrated in which the simulation model is able to respond to the changing circumstances of the physical system by way of an error event which causes the simulation to restart, and continue scheduling based on the new configuration (Beregi, Szaller and KáDáR 2018). It is accepted that the production schedule will be unique, but the permutations could be tested in advance using traditional Discrete Event Simulation methods.

There are many papers which show the method of data capture for a type 3 Digital Twin (Schroeder et al. 2016; Zhuang, Liu and Xiong 2018) although there are no actual implementation examples. The article by Rosen et al points to the decisions an intelligent production system, type 4 Digital Twin, could make in order to maximise throughput, although again, this a theoretical paper with little detail of the artificial intelligence system which could be used (Rosen et al. 2015).

The paper (Omar, Hussain and Wright 1999) shows that it is possible to increase throughput by replacing a traditional transfer process with flexible manufacturing cells with an artificial intelligence control system, but concedes that is at the expense of vastly increasing the working in progress.

The paper (Vachalek et al. 2017) shows the benefits of a digital twin based upon the transfer of data from PLC controlled machines, which is very relevant to Crown. The paper shows theoretical advantages but no physical implementation. A review of literature by Kritzinger shows that most papers where a Digital Twin is the subject refer to type 1 and 2 Digital twin: “only 18 percent of them are really describing a Digital Twin with a bidirectional data transfer” (Kritzinger et al. 2018), with many of them described as “concept” or “case studies”.

Simulation and Discrete Event Simulation have been around for long time. At the inception of the technique, it would have been almost impossible to link the simulation model to the physical system, as the technologies required to do so were decades away, or too expensive to be a practical solution. The simulation model was set up in its own universe, with boundary conditions and data from the real world, but ultimately operating in isolation from the physical system. Most Discrete Event Simulation software companies refer to a Digital Twin, based on their product, mentioned on their website (Austin 2019; Simio LLC 2017a; Simul8 2018) , almost exclusively referring to complex order scheduling, which utilises data from the physical system, but without the closed loop back to the physical system. This is similar to the concept papers for symbiotic simulation in which methodologies were developed to initialise simulation models from real time data from physical systems (H. Aydt et al. 2009; S. Bergmann, S. Stelzer and S. Straßburger 2011).

In order to operate a Digital Twin, the simulation model history must be based upon the physical system as postulated by (Korth, Schwede and Zajac 2018), see Figure 19.

Figure 19 Linking Physical and Simulated Systems(Korth, Schwede and Zajac 2018)

The paper by Yang (YANG, TAN and YOSHIDA, K. & TAKAKUWA, S. 2017) describes a method of linking the simulated system with the real system. In effect the model "history" has become the physical world history. The simulation generated possible future is for decision making only and immediately overwritten with actual history. The current physical state is recorded digitally from an analogue source and stored in a database as a history. In order to use the methodology, the authors created their own simulation code in Visual Basic. It demonstrates the point and also the problem that in order to use the history of the physical system, the simulation code must be designed with this in mind. The well-established commercial codes do not have a very good mechanism to achieve this. Simio, for example, uses an add-on process to generate entities and position them at nodes throughout the model at the beginning of a simulation. This works well and can be easily driven from an external data source but with the exception it is not possible to place entities on a conveyor, between two nodes. As the majority of the work-in-progress, for Crown, is on the conveyors in the model, not being able to initialise conveyor states is major shortcoming.

There are several papers describing the implementation of Industry 4.0 technology in the context of cyber physical systems into industry, for example (YANG, TAN and YOSHIDA, K. & TAKAKUWA, S. 2017)(Lee, Bagheri and Kao 2015)

Figure 20 Implementation levels of Cyber Physical systems (Lee, Bagheri and Kao 2015)

The cost of implementation is always a factor as Weber quotes “Many manufacturing executives also come from engineering backgrounds, meaning they [want] to see hard, measurable evidence about change.”(Weber 2016). It is likely that cost conscious companies similar to Crown will move up the Cyber Physical Systems pyramid at a measured rate, see Figure 20 Implementation levels of Cyber Physical systems (Lee, Bagheri and Kao 2015), upgrading to Industry 4.0 equipment when old equipment comes to the end of life or when a specific need demands it. Each layer of the pyramid will have to be self-sufficient and also forward thinking, i.e. not reliant upon some future upgrade to be worthwhile but also able to be used easily at the next level when the business case permits. A counter view is given by Weber in his myth busting article (Weber 2016) stating that “The reality is that enabling communication or collecting process data from existing equipment is easier and faster than ever before and is more cost-effective than replacing equipment”.

The paper by Rosen (Rosen et al. 2015) draws an important distinction between automation and autonomy. The former, even with more industry 4.0 type sensors, is preprogramed and the computer system is simply following a flowchart to arrive at the decision. The latter involves the human like quality of lateral thinking, given all the sensor information, to arrive at the solution. The paper doesn't address the nature of the intelligence or the cost of implementation versus the extra throughput.

There are relatively few examples of a Digital Twin, at least where the Twin has some kind of Discrete Event Simulation acting as the optimisation engine. In the paper by Kritzinger (Kritzinger et al. 2018), all of the case study literature was based on production planning and control. Most of the Discrete Event Simulation software vendors mention a Digital twin (Austin 2019; Simio LLC 2017a; Simul8 2018), but all in terms of "smart scheduling" but the autonomy is in the hands of the user. This is more closely aligned to the methodology of symbiotic simulation where the physical production line and simulation model are linked allowing for fast experimentation.

In conference proceedings, the Digital Twin was suggested to be in the hype phase of development "few digital twin examples are publicly available for discussion to understand the benefits with even fewer utilising immersive technologies" (Eyre and Freeman 2018)

Figure 21 Hype Cycle with Digital Twin at the peak of expectations (Eyre and Freeman 2018)

4 Results

4.1 Braunstone Discrete Event Simulation Model

The food can production line at Braunstone manufactures 2 piece food cans. The process is similar to beverage can manufacture, in that it is a cupping and redraw process. However, there is no decoration stage and the cans are beaded for extra vacuum strength.

4.1.1 Conceptual Model

The conceptual model, see Figure 22 was constructed from Smartline data and information regarding the capacity of the production line, see appendix2 for detail.

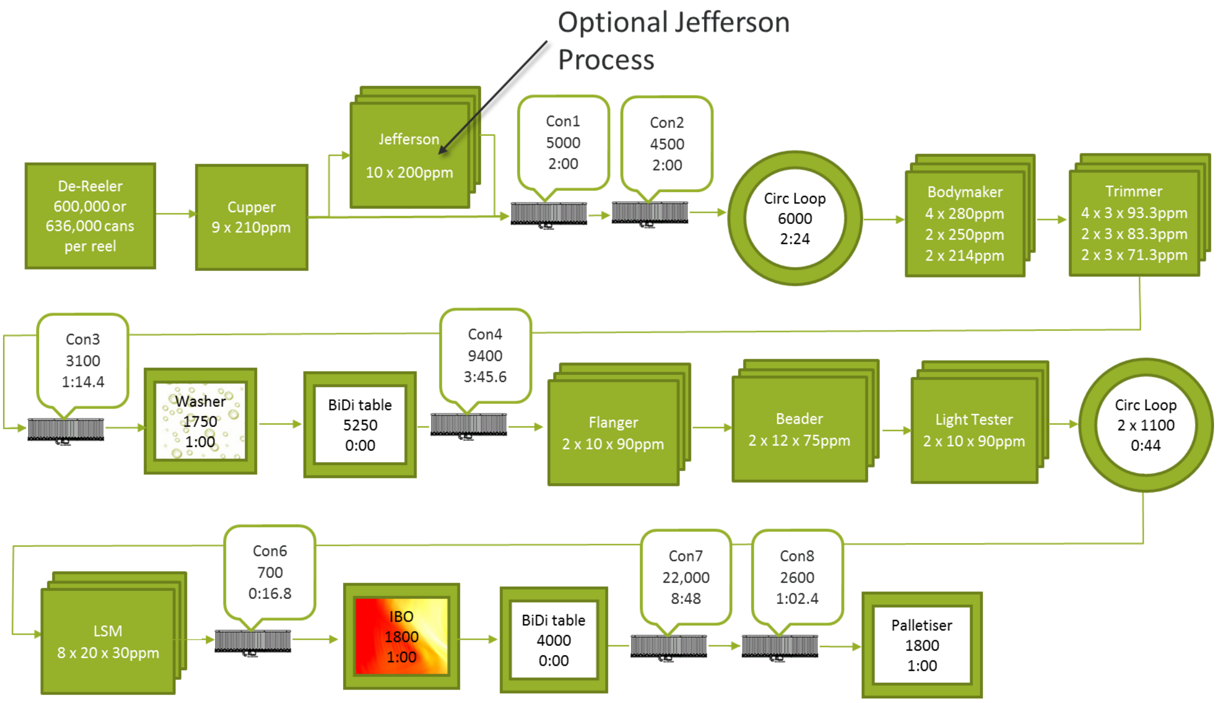


Figure 22 Braunstone Conceptual Model

4.1.2 Baseline Model

The baseline model was constructed, which initially suffered from from extremely poor execution speeds.

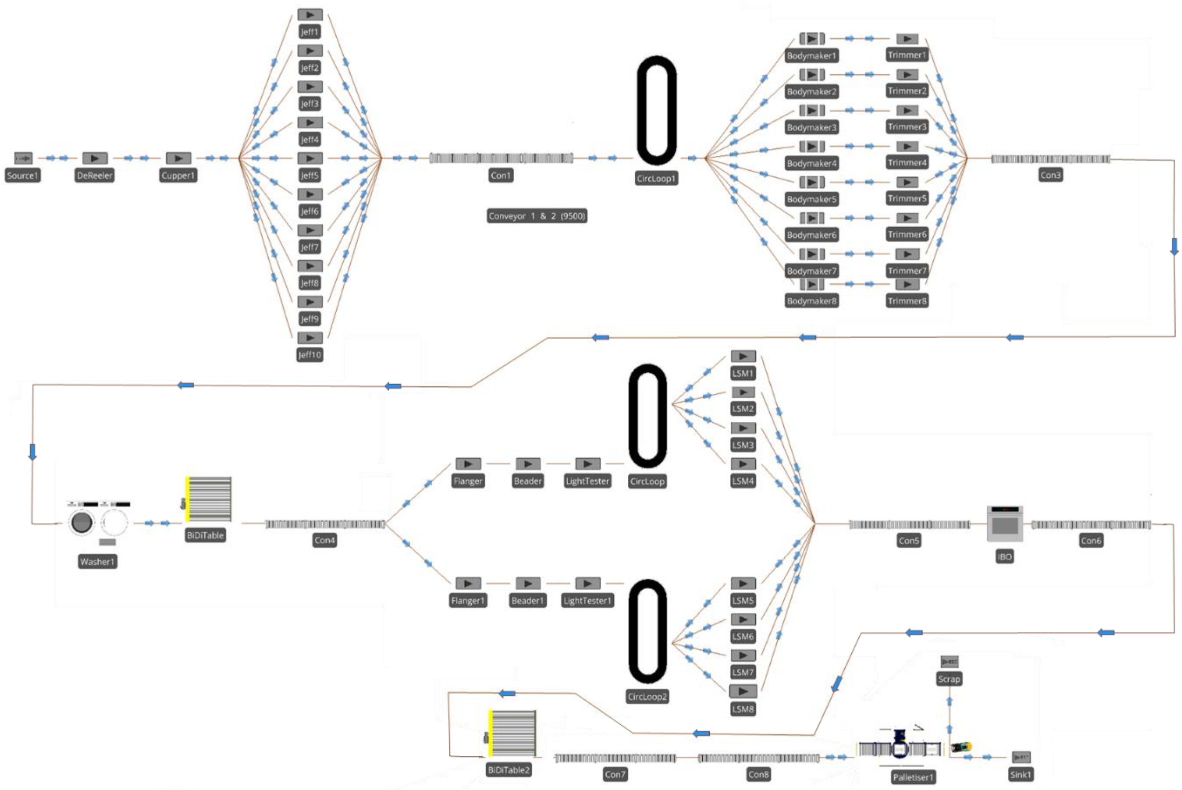


Figure 23 Braunstone Discrete Event Simulation Model

Because the line runs with a large amount of work in progress, on the order of 80,000 cans, the poor performance was related to the large number of entities in the system.

It was decided to scale the model in terms of each model entity representing between 2 and 50 cans. The scale factor would be checked experimentally to ensure accuracy was maintained. The results of analysing the effect of scaling on model accuracy, see Figure 24, show that accuracy is maintained up to a scale of 10, with a difference of less than 2% versus unity. Beyond scale 10, at scale values of 25 and 50, the accuracy of the model is severly compromised at a losses of 7% and 14% respectively.

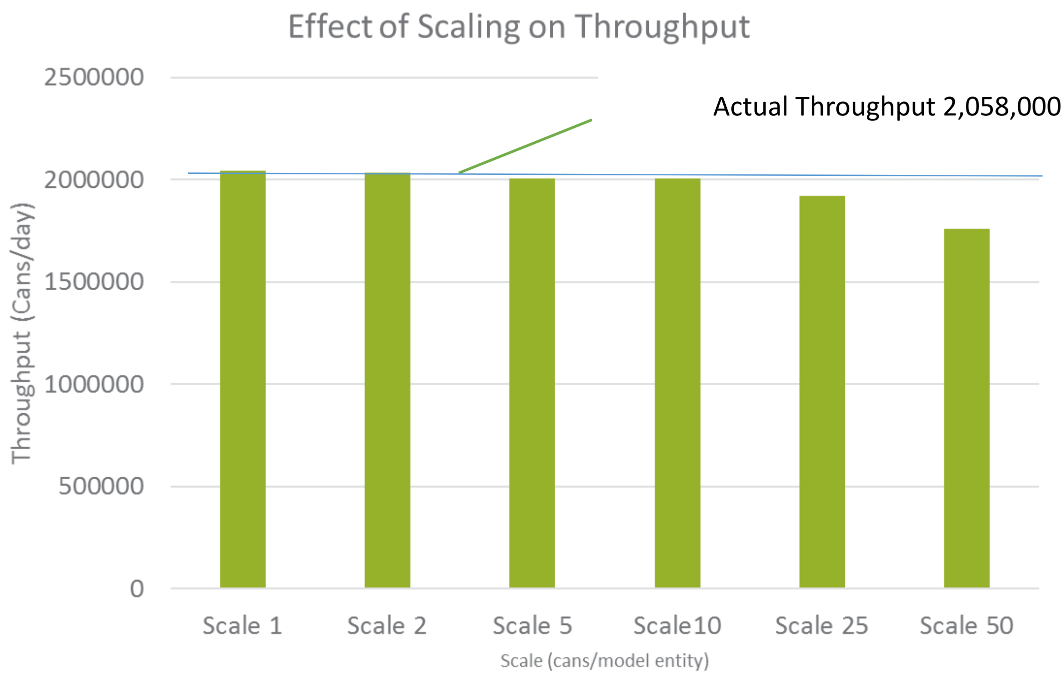


Figure 24 Braunstone Effect of Scaling on Model Accuracy

The system required 5 hours of warmup to populate all the conveyors. The manufacturing process is continous allowing for long simulation runs of 45 days. As the model runs for a relatively long time, then relatively few replications are required, in this case very little deviation over 5 replications, see Figure 25.

The baseline model was validated against the Smartline data see apendix 2.

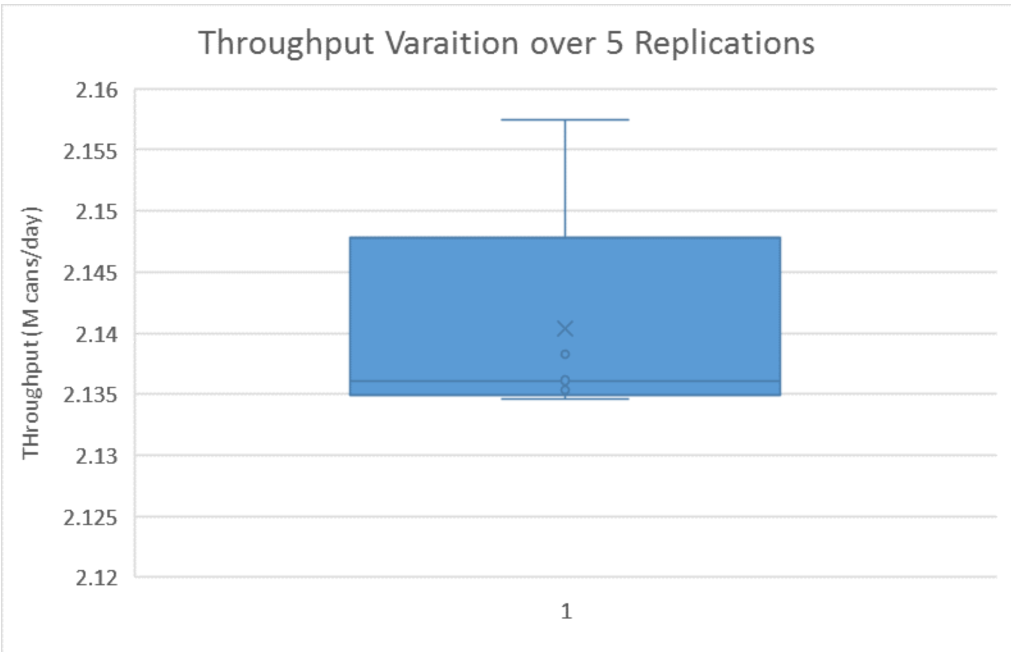


Figure 25 Braunstone Throughput Variation over 5 Replications

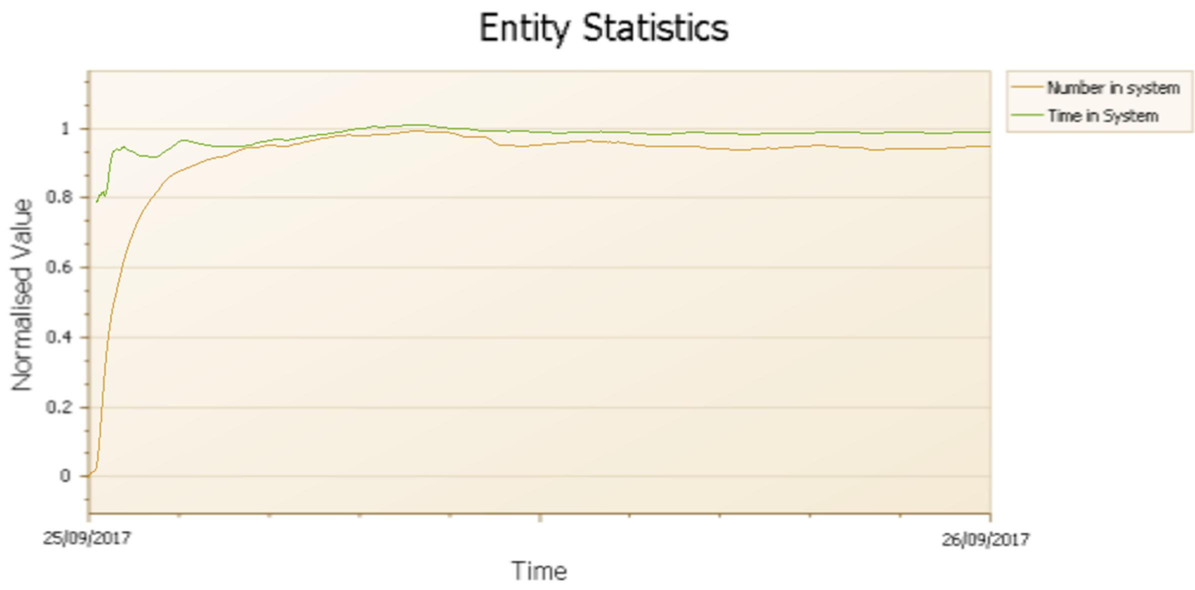


Figure 26 Braunstone Model Warmup Chart

4.1.3 Jefferson Capacity Experiment

The Jefferson machine reforms the cup in order to thin the metal at the base of a 2 piece food can, thus saving a small amount of metal. The machine is expected to increase the yield from a coil of steel from 600,000 cans to 636,000 cans, which is a 5.6% reduction in metal for each can.

The assumptions for failure data were based upon similar expected failures to the Cupper machine at each Jefferson machine. The machines would be expected to work at a rate of 200 cans/minute.

The objective of the Jefferson experiment was to determine how many machines would be required so as to maintain throughput through the line.

The results show that 9 Jefferson machines would maintain the throughput at a level at the baseline level without the additional process. The time saved by having to do fewer changeovers, 3.3 per day down from 3.5 per day, only represents a fraction of the time spent processing and does not give any appreciable extra throughput, see Figure 27 Braunstone Jefferson Capacity vs Throughput.

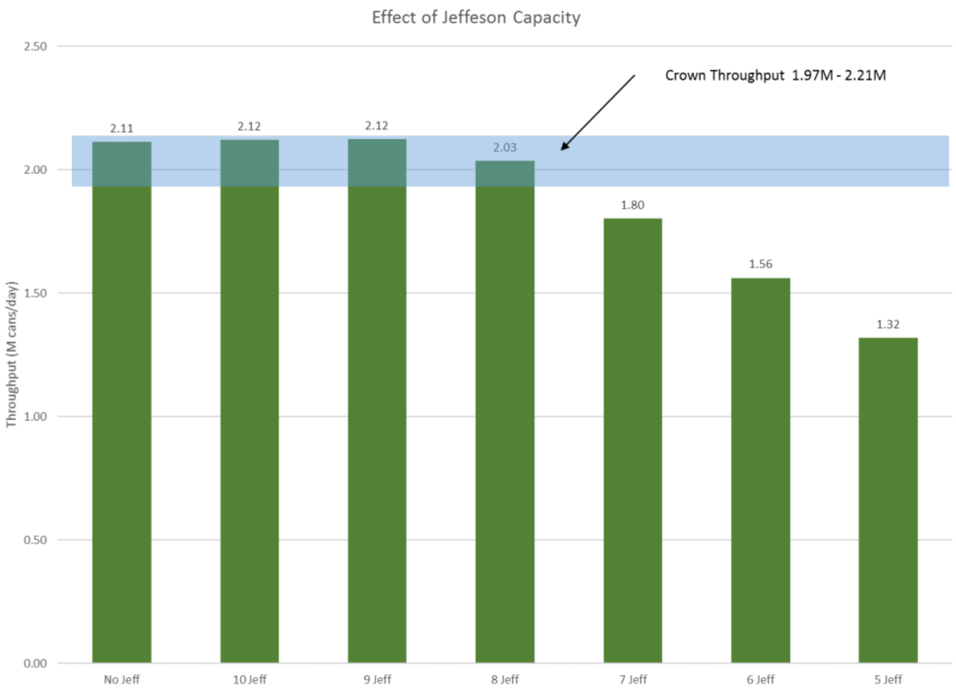


Figure 27 Braunstone Jefferson Capacity vs Throughput

No financial information was given as to the likely cost of each Jefferson machine and with no increase in throughput, the reduction in material in each can alone would have to justify the capital expenditure.

4.2 Custines Discrete Event Simulation Model

The Custines factory has two production lines manufacturing beverage cans. The focus of the study was line 1. The factory is relatively new and production has been ramping up for most of 2017, which meant that there was only a couple of months at the beginning of 2018 with stable data.

4.2.1 Verification

The conceptual model was built with data contained in line layout drawings. Further details are shown in appendix 3. The conceptual model, see Figure 28 Conceptual Model of Custines Line 1, was circulated to line experts for verification, prior to building the Discrete Event Simulation model

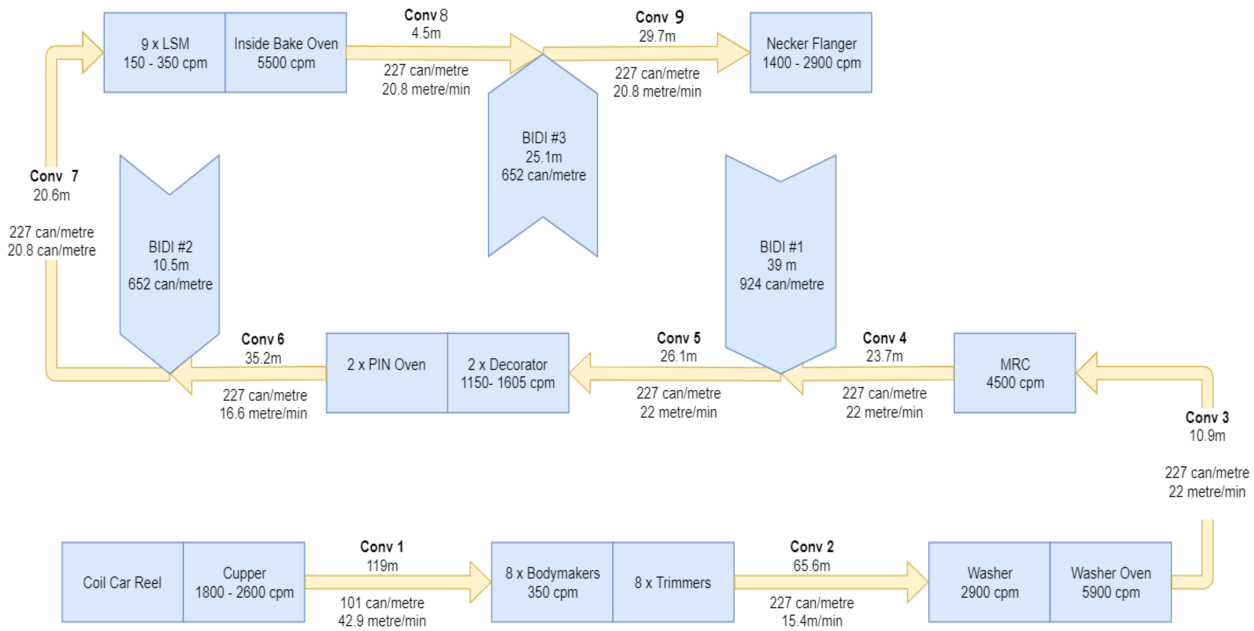


Figure 28 Conceptual Model of Custines Line 1

4.2.2 Description

Crown specifies machine process times in terms of cans per minute; conveyors are specified in terms of capacity, cans per metre, and speed in metres per minute. It was felt by the stakeholders, that the model input should reflect this standard in order that the model may be reused by suitably trained Crown staff.

The individual machines could be represented by two sub models: Larger Server and Small Server. The Large Server represents the washers and ovens with capacities in the thousands of cans. The server would scale the capacity and keep the processing time as specified. The small servers for machines of lower capacities, for example Bodymakers, scales the processing time and keeps the capacity as specified. The large server and small server modifications are detailed in appendix 4 and 5 respectively.

The object oriented nature of Simio allows for the enhancements to servers and conveyors and then utilisation of the modified servers and conveyors in the high level model.

The number of cans in a single line averages ~70,000 peaking at ~99,000. In order to reduce the run time of the model, the entities in the model would be scaled, as in the Braunstone model to, represent a number of cans. The optimum size of scale would be determined by experimentation.

4.2.3 Customisation

4.2.3.1 Small Servers

Small servers are used to represent machines with small capacities (less than 50), specifically the Cupper, Bodymakers, mass rim coater (MRC), lacquer spray machine (LSM) and Necker. The capacity is unchanged but the processing time is scaled by the number of cans represented by a single entity. For example if the model is scaled at 20, then a 60 can per minute process, or 1 per second, will have a scaled process time of 20 seconds.

- All data can be entered in cans per minute
- Daily throughput is tallied.
- Custom state variable to include the machine speed state
- Custom events to trigger speed changes
- Setup state with time and entity count based setup events.

More detail for the small server sub model are contained in appendix 5. The small server sub-model was extensively tested in a small model designed to test all the event triggered functions and output of the server.

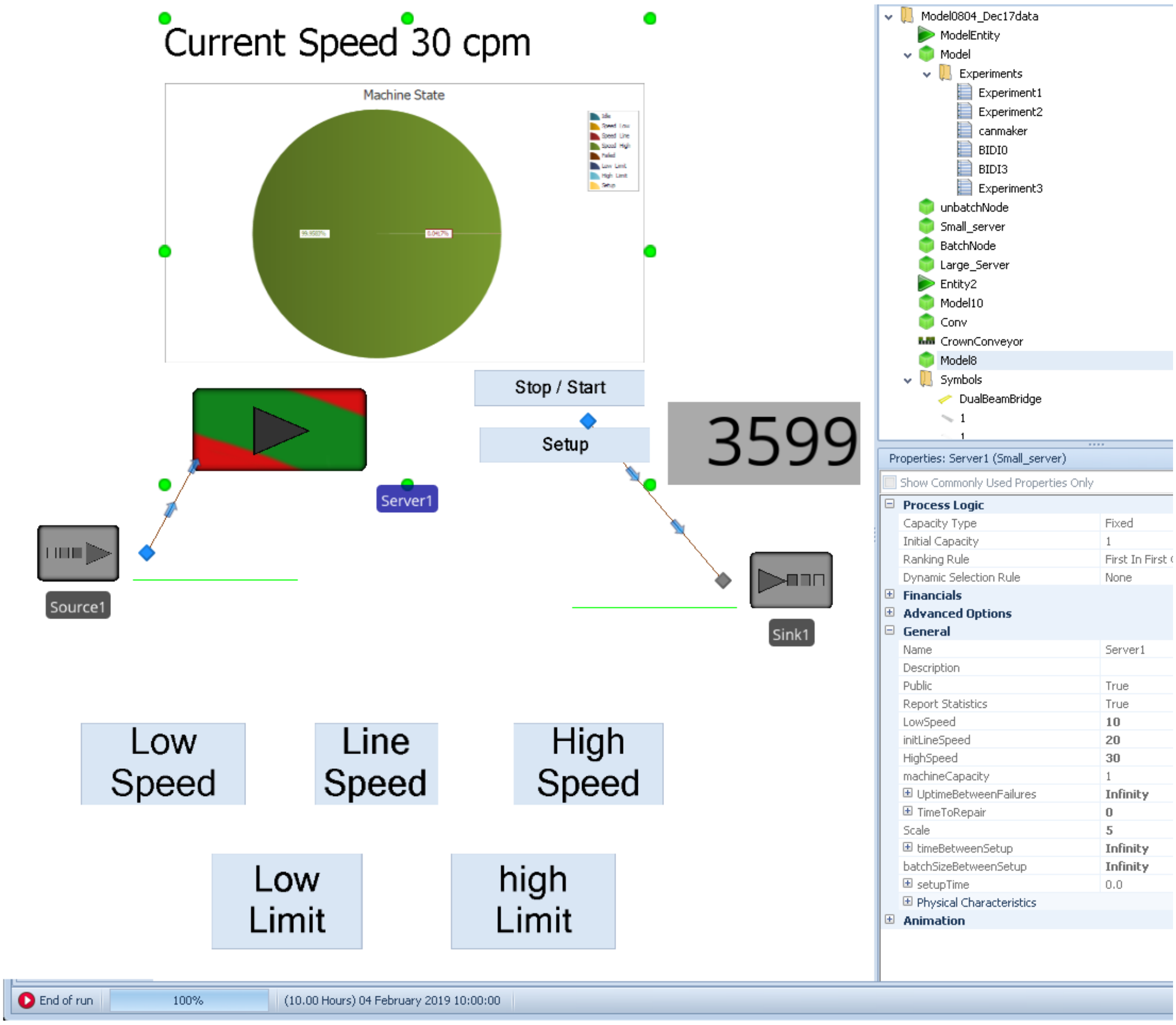


Figure 29 Small Server Test Model

The speed of the servers was easily tested by running the test model for 10 hours and counting the entities through the model. The capacity of the machine is 1 and two scale values tested.

Speed	Count Scale 5	Count scale 20
Low (10 cpm)	6000	6000
Line (20 cpm)	11995	11980
High (30 cpm)	17995	17980

Table 3 Small Server Speed Test Results

The Low limit and high limit events were tested by running the model and clicking the "Low Limit" button and accumulating 1 hour of stoppage for the 10 hour test. At the end, the machine state status chart showed that Low limit accounted for 10% of the total. The "Time to Repair" and "UptimeBetweenFailures" was set at a fixed values in order to test the down time accumulated.

More detail about the Small Server modifications is contained in appendix 4.

4.2.3.2 Large Servers

Large Serve servers are used to represent machines with larger capacities (More than 50), specifically the washer, washer oven, and Inside bake oven (IBO). The processing time is unchanged but the capacity is scaled by the number of cans represented by one entity. Large servers are used in places where further scaling of conveyors is present. For example if the model is scaled at 200, then a server with capacity 10,000 will be scaled to 50 entities.

- All data can be entered in cans per minute
- Daily throughput is tallied.
- Custom state variable to include the machine speed state
- Custom events to trigger speed changes
- Setup state with time and entity count based setup events.

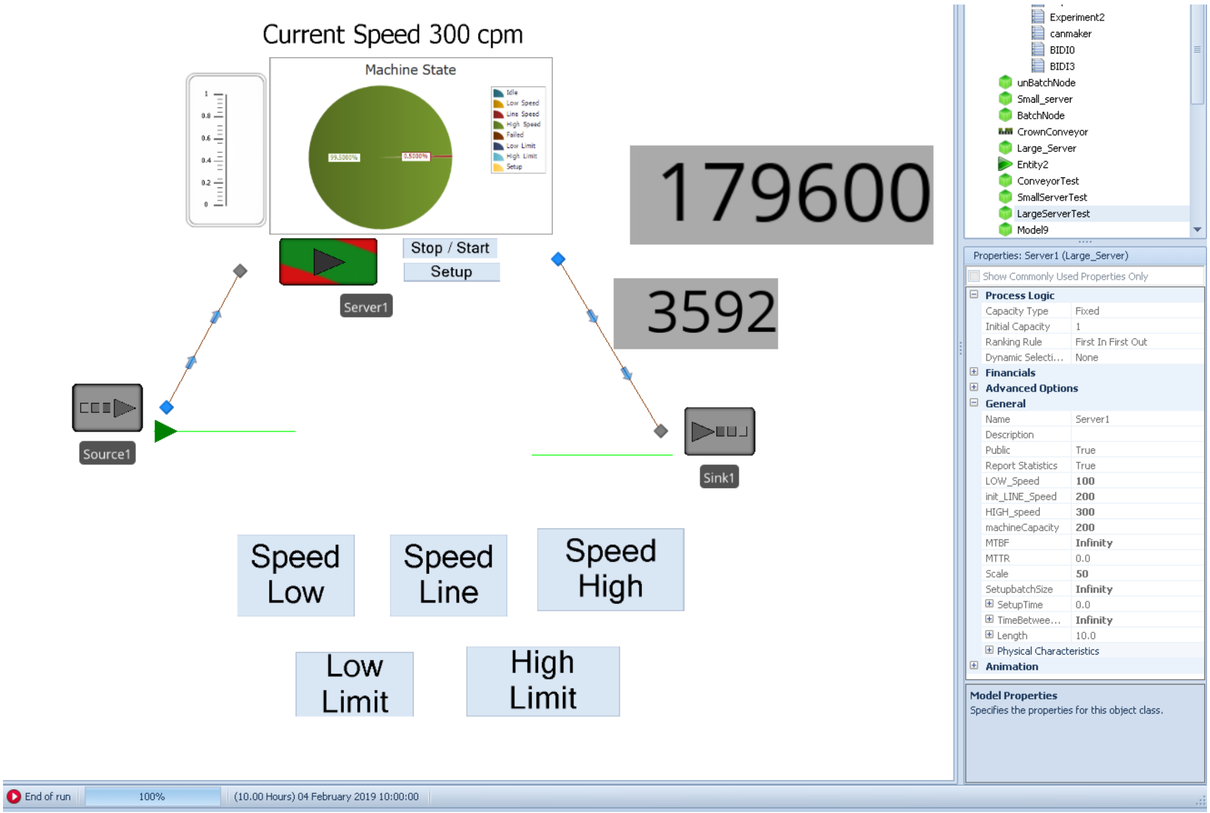


Figure 30 Large Server Test Model

More detail about the Large Server modifications is contained in appendix 4.

The speed of the server was easily tested by running the test model for 10 hours and counting the entities through the model. The test model the entities are pulled by the system rather than generated at fixed intervals. As an entity is destroyed at the sink, the event triggers the creation of a new entity at the source. The machine capacity set 200 and two levels of scale tested. Table 4 Large

Server Test Results, shows the throughput of the server to be within expectations. The anomalous result for low speed, where more entities are counted than expected, is explained by the fact that the servers all start at "Line" speed and in the time taken to press the button to set to "Low" speed, several entities are already processed at "Line" speed.

Speed	Count Scale 50	Count scale 200
Low (100 cpm)	60200	60000
Line (200 cpm)	119800	119800
High (300 cpm)	179800	179800

Table 4 Large Server Test Results

The Low limit and high limit events were tested by running the model and clicking the "Low Limit" button and accumulating 1 hour of stoppage for the 10 hour test. At the end, the machine state status chart showed that Low limit accounted for 10% of the total.

4.2.3.3 Conveyors

The conveyor was a sub-class rather than a sub-model in order to preserve the functionality of the conveyor object. The conveyor needed to be able to work between nodes and have extra vertices to represent the layout visually. It also need to work as a conveyor allowing entities to accumulate, if necessary. The animated graphical display of the conveyor object would also be retained.

The Simio conveyor object does not allow a variable to drive the logical length so the approach used was to scale the speed of the conveyor so as to give the correct travel time on the conveyor and scale the size of entities as they enter the conveyor to give the correct number of entities per unit length.

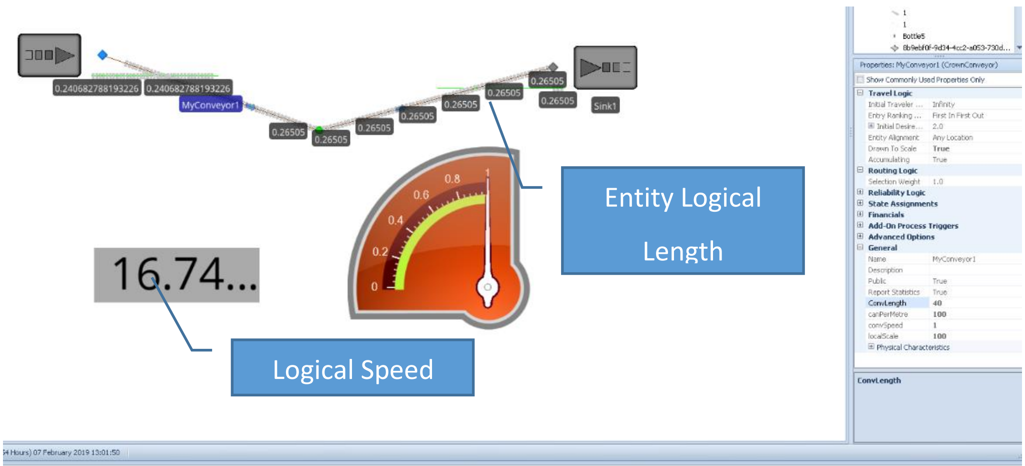


Figure 31 CrownConveyor Test Model

The conveyor was tested to ensure that the graphical speed and entity length was calculated correctly. The conveyor was tested to ensure that the entities accumulated correctly and that the travel time was correct with respect to the logical length and speed, for example with ConvLength set to 60 (metres) and convSpeed set to 1 (metre/min) then the travel time of the first entity was recorded as 1 hour. Figure 31 CrownConveyor Test Model shows the entities displaying their graphical length and a textbox reporting the graphical conveyor speed.

More detail about the Crown Conveyor modifications is contained in appendix 7.

4.2.3.4 Batch & Un-batch Nodes

The batch and un-batch nodes allow for temporary upscaling on mass conveyors and large servers whilst maintaining a relatively small scale through small servers. They are placed around the large servers and conveyors to reduce the number of active elements in the model.

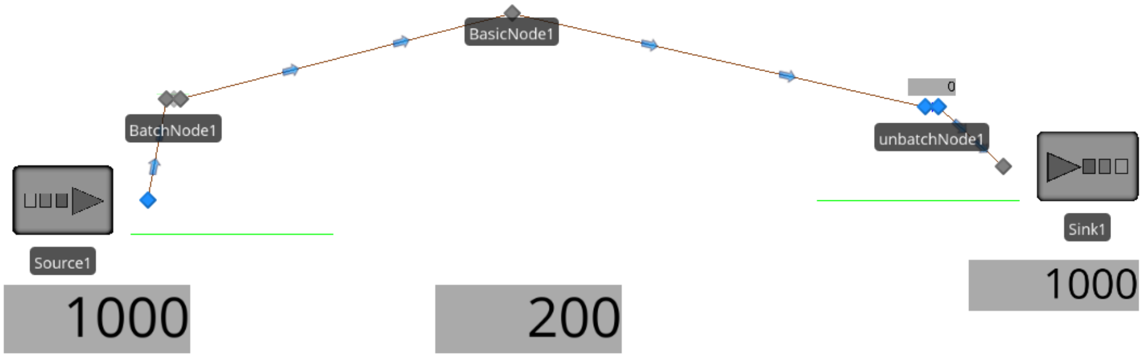


Figure 32 Batch / Un-batch Node Test Model

The batch and un-batch nodes were tested with a sub model which released a known number of elements and counted the elements in and out and at a station between the batching and un-batching nodes. Figure 32 Batch / Un-batch Node Test Model shows the test model with a scale value of 5.

More detail about the Batch and Un-batch node modifications is contained in appendix 6.

4.2.3.5 *Discrete Event Simulation Model*

The Custines model was assembled from data in Smartline for the failure data and the conceptual model, Figure 28 Conceptual Model of Custines Line 1.

The main key performance indicator (KPI) was the throughput of the system.

The model was constructed entirely from the sub models and sub-classed components as described in section 4.2.3 to create the complete line model, see Figure 33.

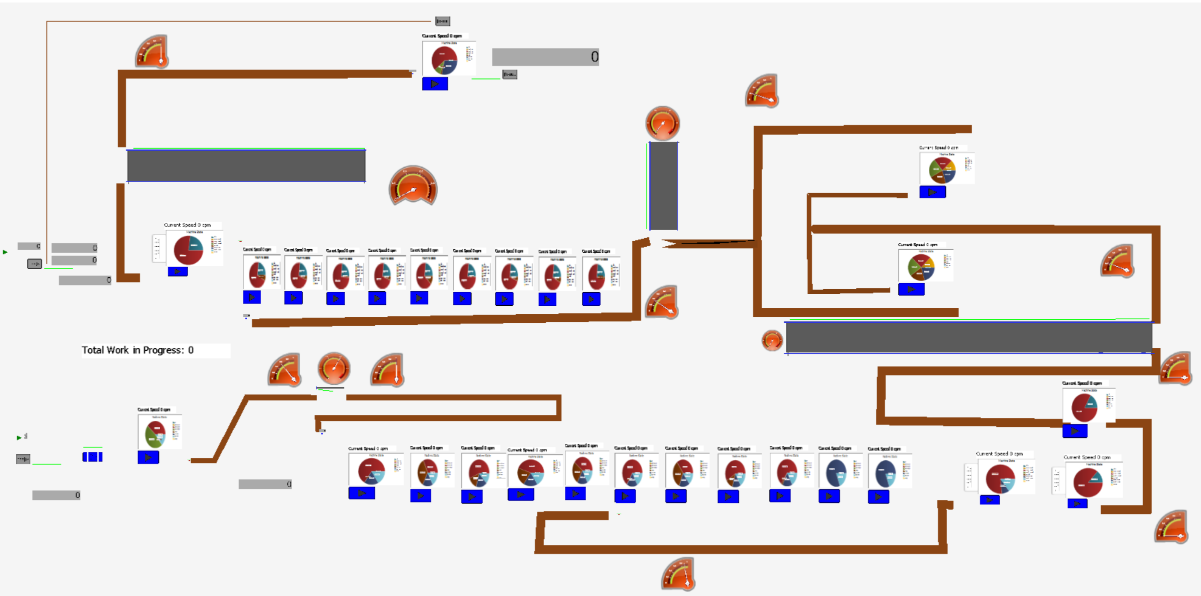


Figure 33 Graphical Display of Discrete Event Simulation Model of Custines Line 1

In order to demonstrate that the model could be driven from an external data source, the machine and conveyor data were stored in an Excel spreadsheet file, see Table 5 Excel Input Data for Discrete Event Simulation Model. The model can be linked to a number of different data sources which become selectable as a model parameter.

The output of the model was via csv file for analysis in Excel which required manipulation in order to generate graphical output, for example machine utilisation stacked bar charts.

Custines Machine Data

Name	Capacity	ProcessRateLow	ProcessRateLINE	ProcessRateHIGH	MTBF	MTTR
Cupper	1	1800	2600	2700	random.Exponential(1.311562)	random.Exponential(0.033613)
Canmaker	1	250	350	350	random.Exponential(0.361732)	random.Exponential(0.093403)
Washer	16000	2900	2900	2900	random.Exponential(16.617009)	random.Exponential(0.148278)
Washer Oven	16000	5900	5900	5900	random.Exponential(27.149182)	random.Exponential(0.670849)
MRC	1	4500	4500	4500	random.Exponential(14.336878)	random.Exponential(0.218201)
Decorator	1	1150	1500	1605	random.Exponential(0.002640)	random.Exponential(0.000831)
LSM	1	150	350	350	random.Exponential(0.282553)	random.Exponential(0.023488)
Inside Bake Oven	12600	5500	5500	5500	Infinity	0
Necker Flanger	17	1400	2600	2900	random.Exponential(0.188317)	random.Exponential(0.016562)

Index	Conveyor	ConvLength	ConvSpeed	canPerMetre	rate in canp per minute	time
1	Conveyor1a	30	42.9	101		0.70
2	Conveyor1b	89	42.9	101	4333	2.07
3	Conveyor2	65.5	15.4	227	3496	4.25
4	Conveyor3	10.9	22	227	4994	0.50
5	Conveyor4	23.7	22	227	4994	1.08
6	Conveyor5	26.1	22	227	4994	1.19
7	Conveyor6a	35.2	16.6	227	3768	2.12
8	Conveyor6b	35.2	16.6	227	3768	2.12
9	Conveyor7	20.6	20.8	227	4722	0.99

Table 5 Excel Input Data for Discrete Event Simulation Model

4.2.4 Baseline Model

The model scale sets the scale of the model through the single entity machines (Cupper, Bodymaker, and LSM machines). The conveyors and other machines run at a higher scale which is the product of the Model Scale and Conveyor Scale.

Following from the Braunstone case study, I made the assumption that unit scaling, i.e. one model entity represents one can, would prove too time consuming. I did repeat the experiment to ensure that the Custines model was not more sensitive to scaling than the Braunstone model

The Custines scale experiment showed that, in terms of throughput, the model is not sensitive to the Model Scale and Conveyor used. It also showed that the Conveyor Scale was not as large an influence on execution speed. The Model Scale has a large effect on execution time, see Table 6 Scale Experiment Results and Figure 34 Custines Effect of Model Scale on Accuracy.

In order to minimise the risk of losing model accuracy, The Model Scale of 20 and Conveyor Scale of 10 were used for further experimentation. The model was validated using January 2018 data where the throughput was 2.67M cans.

Model Scale	Conveyor Scale	Output (millions cans per day)	Wall Time (s)
Actual		2.670	-
5	10	2.540	8818.3
10	10	2.472	4147.3
20	10	2.595	2259.7
40	5	2.593	1184.3
5	20	2.619	7251.6
5	40	2.511	6245.1

Table 6 Scale Experiment Results

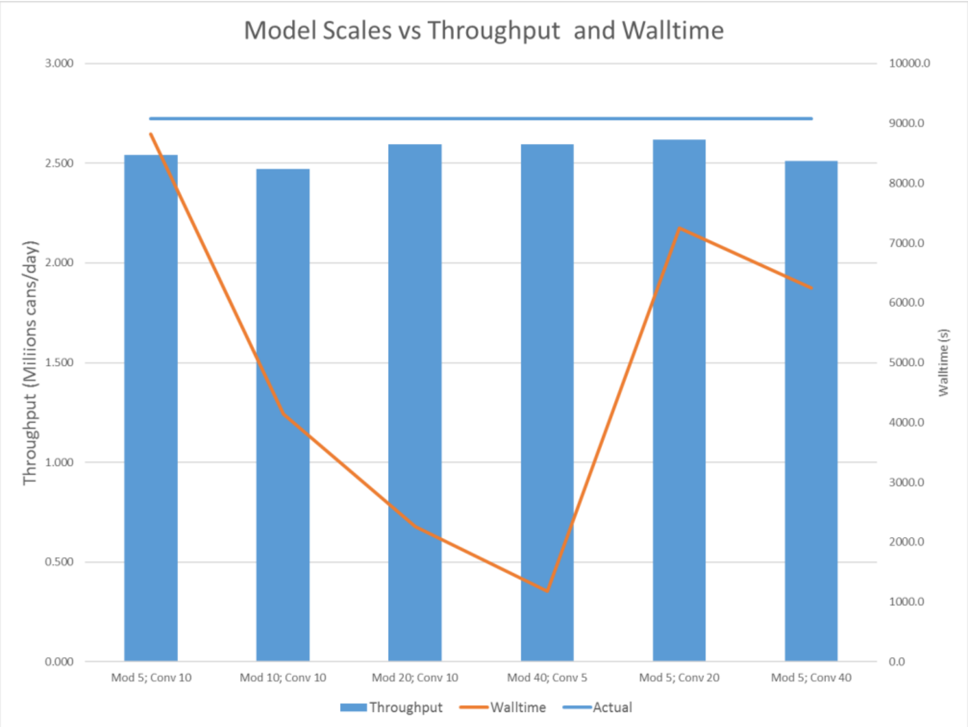


Figure 34 Custines Effect of Model Scale on Accuracy

The warmup of the model was determined by measuring the entity time in system to become stable, see Figure 35. The number of entities in the system metric was disregarded as being too volatile.

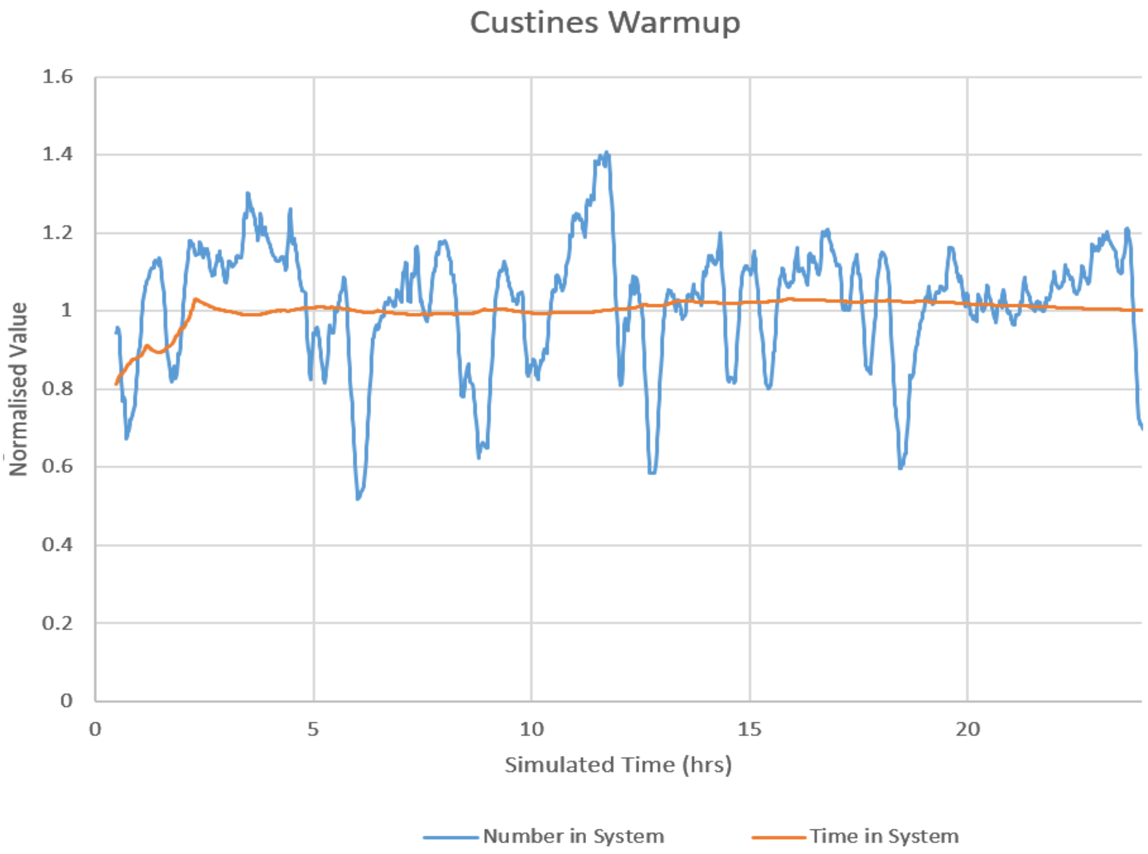


Figure 35 Custines Model Warmup

The variation in throughput of the baseline model was compared with Smartline output from January 2018. The similarity in the variation, see Figure 36, was also good evidence that the baseline model was behaving in a similar manor to the real system.

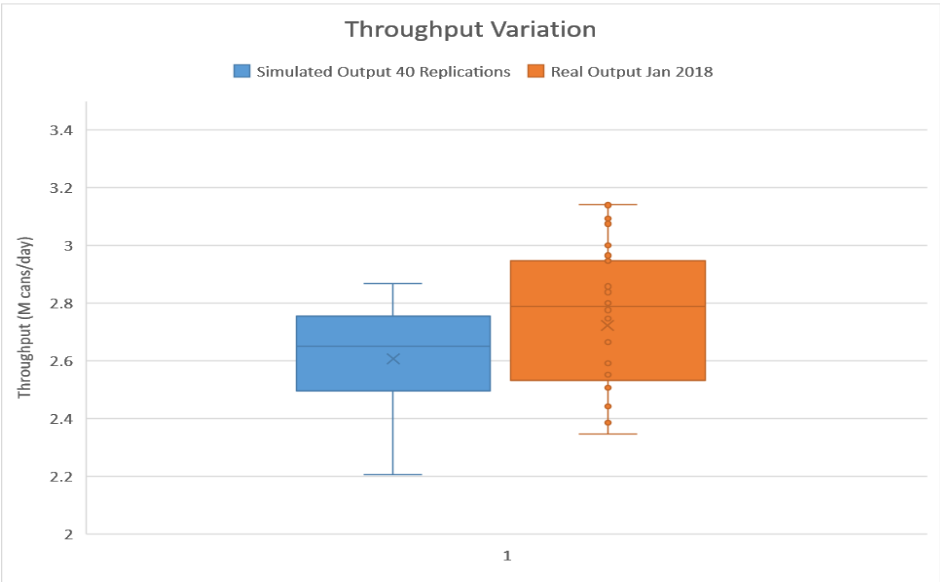


Figure 36 Custines Throughput Variation

The model parameters for the baseline model and experimentation were fixed as per Table 7.

Model Scale	20
Conveyor Scale	5
Warmup	0.25 days of model time
Simulation period	5 days of model time
Replications	40

Table 7 Custines Model Parameters

4.2.5 BiDi3 Capacity Experiment

BiDi 3 is the buffer table immediately following the inside bake oven. Most (76%) of its 16300 can capacity is dedicated as an emergency storage in order to empty the oven in the event of a failure blocking the free movement of cans onwards from the BIDI. 24% of the BIDI is used as a buffer, but as the BIDI continues to fill, the entry to the LSM machines is stopped and the oven empties on to the BIDI. It was hoped that by increasing the capacity of the BIDI, and allowing the LSM machines to continue to operate for a little extra time, then there might be a corresponding increase in throughput.

The experiment was performed with the number of Bodymakers set to 8 as in the current configuration, and also 11 so as to not unduly starve the back end of the line.

Figure 37, shows that increasing the capacity of the BiDi does not increase throughput. Even increasing the number of Bodymakers from 8 to 11 in order to put extra pressure on to the back end of the line, whist increasing total throughput, does not increase the effectiveness of the Bidi as a buffer.

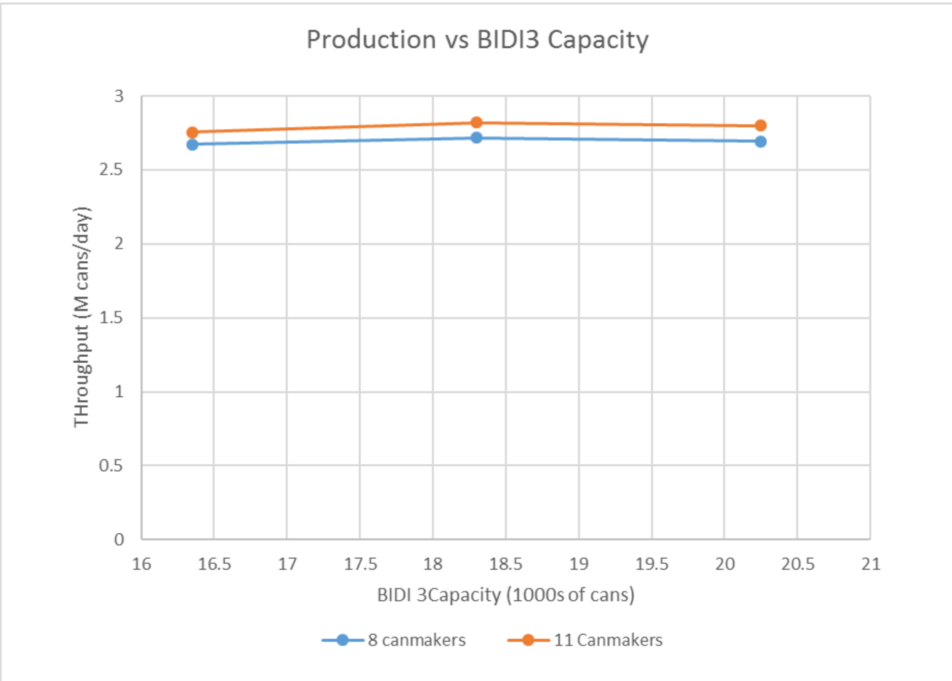


Figure 37 Custines: Effect of BiDi 3 Capacity on Throughput

4.2.6 Bodymaker Experiment

The model was run with Jan18 and Feb 18 data. The output for both experiments shows that the output plateaus at 9 Bodymakers, see Figure 38 Custines Effect of Number of Canmakers on Output.

The output for the January data shows a slight dip in production for 10 Bodymakers compared to 9 Bodymakers. Although this is likely to be a shortcoming in the number of replications or run time, the case for a 10th Bodymaker is certainly disproved

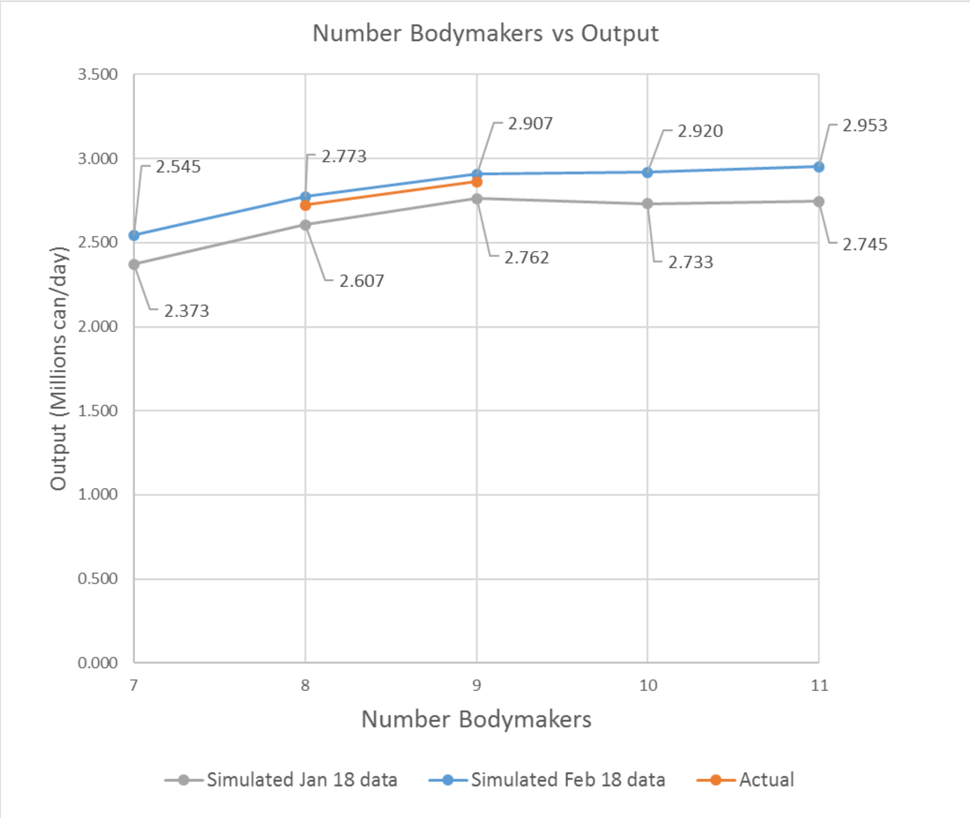


Figure 38 Custines Effect of Number of Canmakers on Output

At a contribution of $15 per 1000 cans, the payback period for the 9th Bodymaker is between 215 and 250 days. Similarly, the payback for the 10th Bodymaker is not guaranteed and is likely to be greater than 2600 days.

4.2.7 BIDI0 Experiment

Bidi0 is a theoretical buffer which could be placed between the Cupper and the Bodymaker group. The theory being tested is that the buffer could hold enough cans to enable the Bodymaker group to continue working during periods of Cupper downtime (breakdowns and changeovers).

The results show that even a massive buffer table would not have an appreciable effect on throughput, see Figure 39 Custines Effect of Additional Buffer between Cupper and Bodymakers.

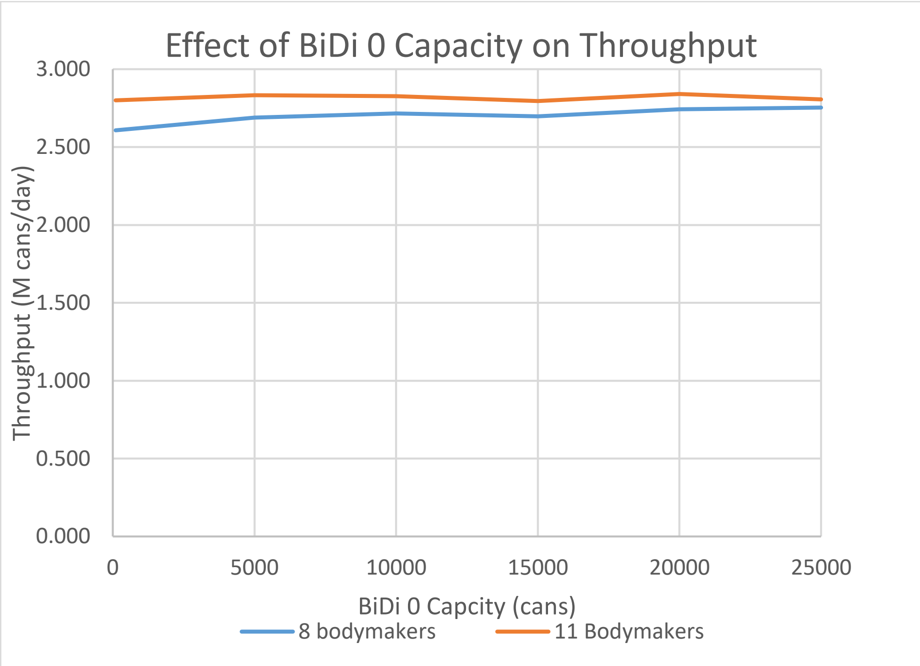


Figure 39 Custines Effect of Additional Buffer between Cupper and Bodymakers

4.2.8 Line Controls Experiment

The line controls are implemented to protect can damage in four ways:

1. Stopping machines where the output conveyor is already full. This is the most rudimentary aspect and is implemented on all conveyors and sub conveyor systems. The Custines model did not include this logic as this behaviour is already captured in Discrete Event Simulation software.
2. Ensuring air table conveyors have a minimum capacity to reduce collision speed of cans. This was implemented on conveyor 1, between the Cupper and the Bodymaker group.
3. Stopping entry into conveyor ovens when there is no escape capacity remaining. This is to ensure that the ovens can always empty their contents, normally onto a BiDi table. This logic was implemented at the Washer and IBO where only a fraction of the BiDi following each was available as a buffer.
4. Modulating machine speed to reduce stop/start behaviour. This was implemented at the Decorator matching the speed of the Bodymaker group and the Necker matching the output of the LSM group

The baseline model included the line control logic above. The controls were disabled in order to measure the dampening effect of the controls on throughput.

Running the model without line controls the increases throughput by 6.7%, see Figure 40 Custines Effect of removing Line Controls.

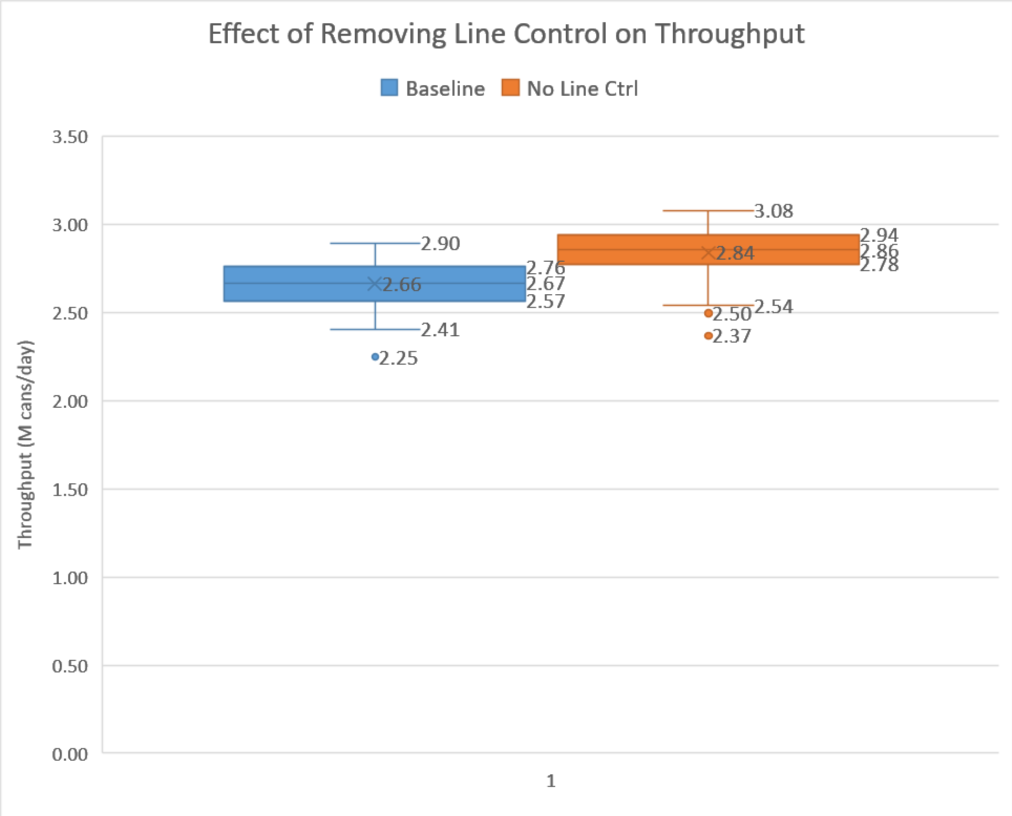


Figure 40 Custines Effect of removing Line Controls

The experiment demonstrated that there was a small amount of scope for further line control experimentation.

The second experiment involved adjusting the Line Control parameter for BiDi 3 so as to utilise more of its capacity as a buffer between the IBO and the Necker. As cans cannot spend any more time in the IBO as heat spoils the decoration, an extra sink was placed after the IBO to dispose of cans which cannot transfer on to BiDi3. The theory being tested was that the extra buffer capacity would increase throughput with only a minor increase in spoilage.

Figure 41 shows the effect of this more aggressive Line Control strategy in terms of extra spoilage produced and throughput of the system. The logic works as expected in that as more of BiDi3 is used for buffering, there is less space available to empty the IBO, leading to an increase in spoilage, but there is no extra throughput gained from the extra buffer capacity, thereby disproving the theory.

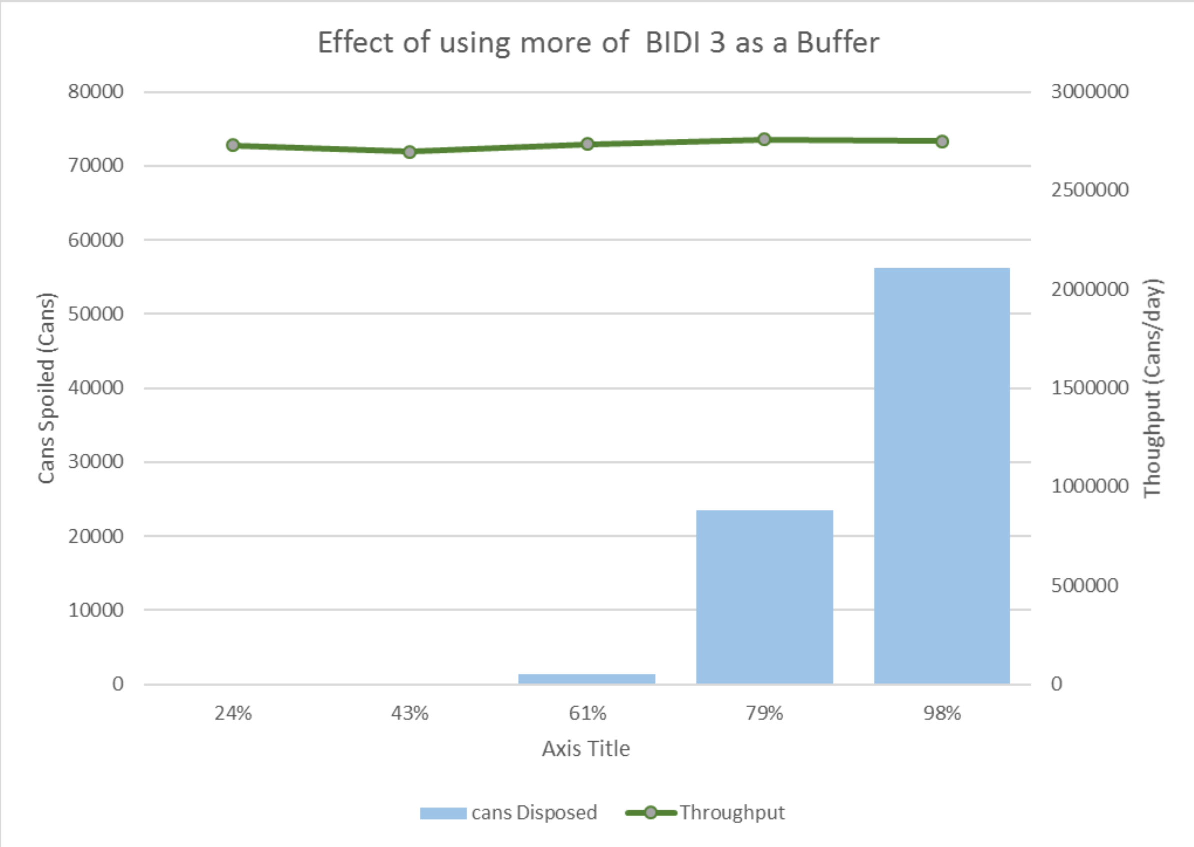


Figure 41 Effect of more Aggressive Line Control Strategy for BiDi *3*

4.2.9 Conveyor Initialisation

There is a challenge with regard to initialisation of work in progress on to Discrete Event Simulation conveyor objects in that most Discrete Event Simulation only allow for model entities to be initialised onto stations or nodes. There is a workaround which involves placing the entities at the node at the head of the conveyor and then temporarily increasing the speed of the conveyor so that the entities accumulate at the end of the conveyor, see the workflow in Figure 42.

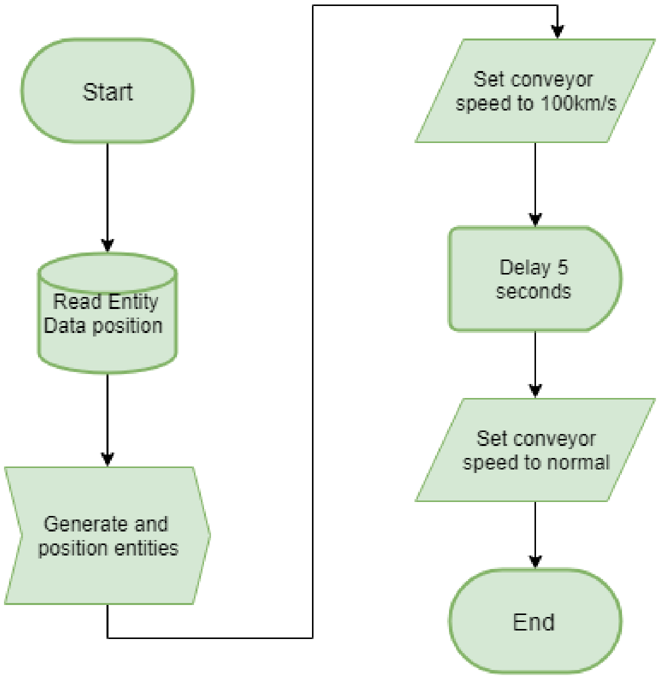


Figure 42 Simio Conveyor Initialisation Technique

The technique was not used for an experimental case study, but demonstrated as working in a small model shown in Figure 43. The technique is not perfect as all the entities are initialised accumulating to the end of the conveyor, but can be driven from a data source and is claimed to be more accurate than starving the processes following the conveyor while the entities traverse the conveyor and the model settles into a steady state.

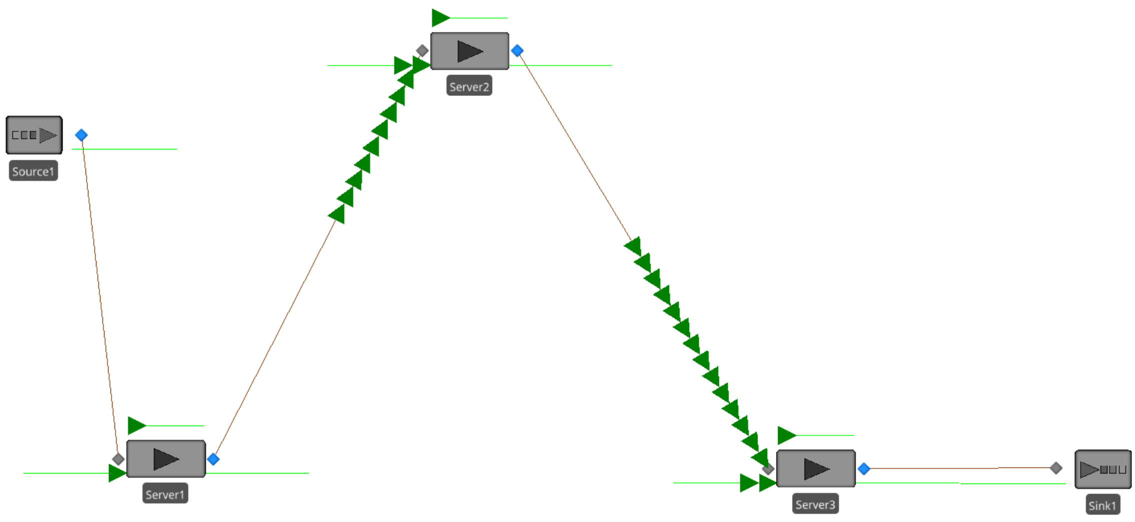


Figure 43 Simio Conveyor Initialised

4.3 Interviews with Crown Experts

4.3.1 Order Control

Crown factories uses one of two systems, SAP and JD Edwards, depending on the geographic location and the age of the facility. New factories tend to use SAP.

At this time, the orders are entered in to the order control system, SAP of JD Edwards, and the production planning is done in a semi-automated way with paper instructions sent to the production staff, daily and weekly, in order to set up the line and complete the order.

Both SAP and JD Edwards systems have a WEB API which could be utilised to push the production orders to a computer based production planning system.

4.3.2 Line Control System

The line controls system can control the speed of machines based on conveyor current capacity thresholds and state of servers up or downstream. The Line Controls system is a PLC based control for the machines and conveyors on each production line. The purpose of the PLC based Line Control system has many objectives:

- To prevent can damage by ensuring that machines do not output on to already full conveyors
- To prevent can damage by ensuring air table conveyors maintain a minimum capacity to reduce collision speed of cans.

- Ensure conveyor ovens always have enough empty space in which to empty out on to in case of breakdown.
- Modulate the speed of machines in order to prevent stop/start behaviour which leads to greater spoilage as the machine produces poor, often unacceptable results during the warm up phase. This warm up phase can also increase the likelihood of machine downtime due to jams.

The system attempts to control the conveyors based on the can density of cans on the conveyors. It detects conveyor capacity using light sensors placed at intervals on the conveyor length, see Figure 44. The system can control both the motors and can modulate the speed of machines, depending on the algorithm in the PLC program.

In theory, the systems stops a section of conveyor when it is full and the section ahead of it is stationery. For example in Figure 44 the motor controlling the middle section, M2, will continue to run so long as motor M1 is running and the sensor S2 shows as not full.

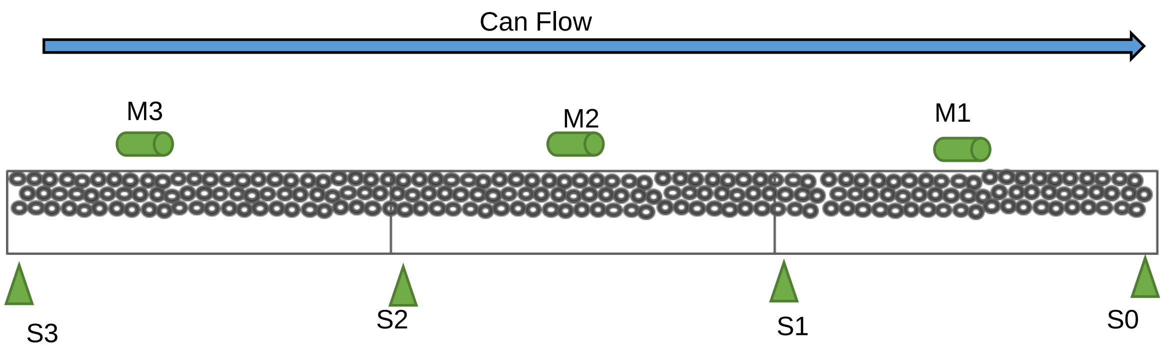


Figure 44 Crown Line Controls Operating Normally

However, it is accepted that the system can be fooled in to stopping conveyors prematurely, see Figure 45 Line Controls Stopped. The Line Controls system has stopped the middle section even though there are voids on the conveyor.

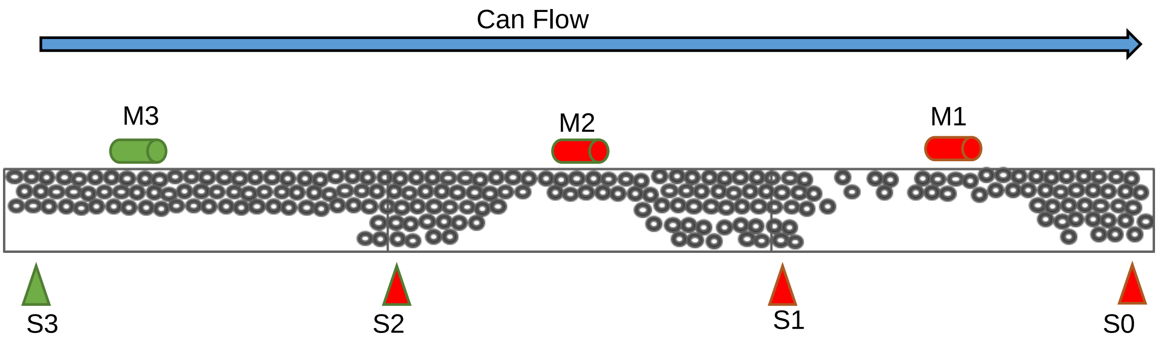


Figure 45 Line Controls Stopped

If Crown were to introduce a more intelligent method of estimating the contents of estimating the number of cans on a conveyor, then gaps in the flow could be eliminated. There are a couple of methods to accomplish this, which initially would not require much of an investment:

The first method involves utilising the can counts at the single process machines (Cupper, Bodymaker, Decorator, LSM, Necker and Palletiser). The system could estimate losses due for spoilage and also it could be intelligently reset every time the conveyor empties or reaches a threshold maximum. On production lines that run regular decoration changes, the conveyors are meticulously emptied at the end of each batch and therefore provide a naturally occurring opportunity to reset the conveyor counts from the decorator onwards. Additionally, the line control build-back sensors could feed into the algorithm.

It is accepted that the PLC program, whilst being extremely reliable, is very opaque when compared to systems such as Smartline. The conceptual drawings of the control logic are absent and PLC programs, although commented, are often difficult to understand.

It is Crown's policy to retain control of the overall conveyor control, rather than place in the hands of subcontracted agents or suppliers of the conveying equipment but by maintaining the PLC based system does not allow for easy experimentation of alternate control strategies.

4.3.3 Line Layout

The line layout of a facility is either stored as electronic or paper drawing or depending on the age of the facility. The oldest facilities do not have drawings of any format, although block diagrams are a suitable alternative.

The more modern facilities have CAD documents showing the length and width of conveyors and placement of Line Control build-back sensors. An example of line CAD drawing is contained in appendix 3.

The older facilities the drawings, where drawings simply do not exist may have produced block diagrams, see Figure 46, which although containing less information than CAD, has enough detail to reconstruct the line for a Digital Twin.

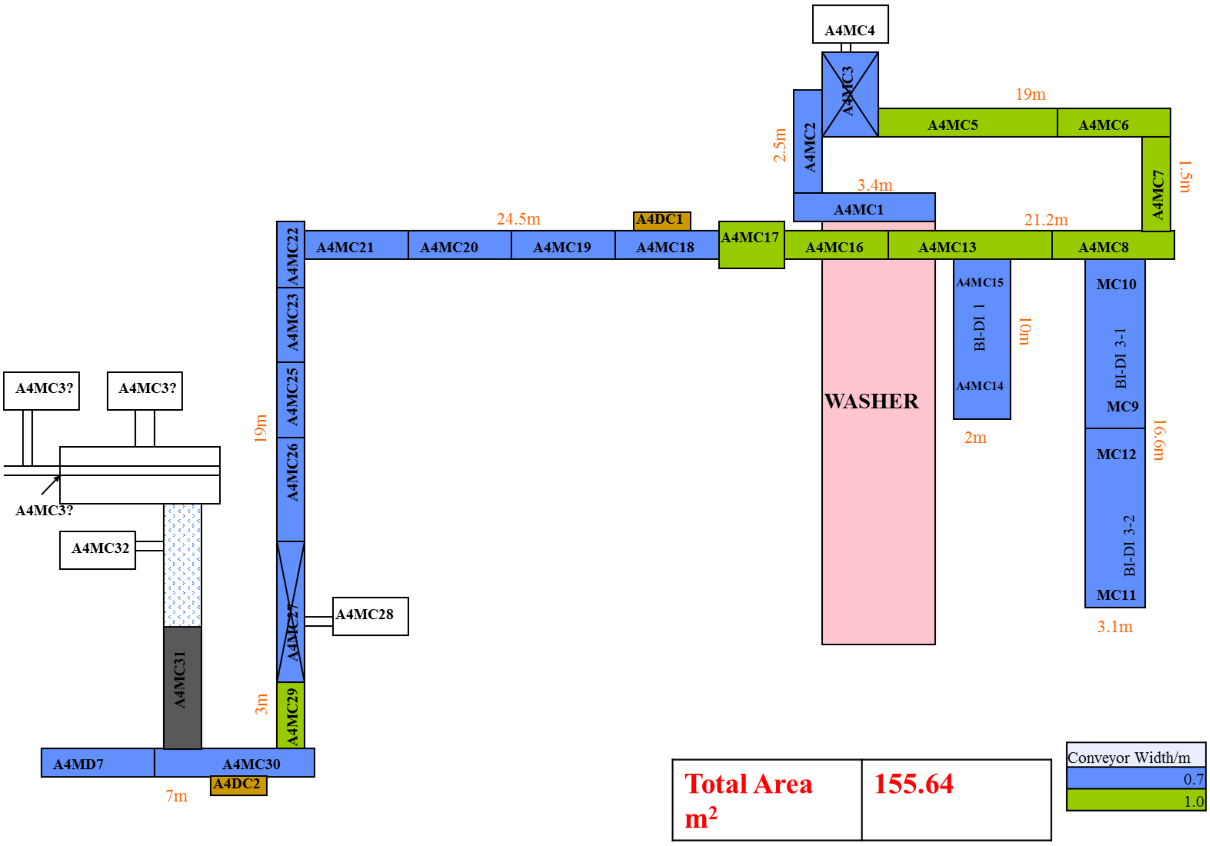


Figure 46 Block Diagram of Section of Production Line

4.3.4 Partial Pallet

Crown receives order in terms of complete pallets of cans. Some customers will accept over or under production, but increasingly customer requirements are for complete pallets only. Crown has to either destroy over production of an order or store as part pallets in the warehouse, both of which add to spoilage rates or storage costs.

Crown has a can counting system implemented as a PLC program, in a one of its beverage factories. The system is used to stop the decorator when the work in progress (WIP) on the conveyor system is enough to complete an order and prevent over production. It estimates the WIP by counting production at the decorator and deducting cans counted on to pallets by the palletiser machine, taking into account spoilage. The spoilage calculation is based on historic completed production runs.

It is a successful implementation and could be repeated at other Crown facilities but it has some drawbacks:

- Increased licensing cost for extra PLC input output screens (HMI).
- Not directly repeatable. Due to the diverse nature of PLC program structure.
- Paper system. There is no direct link to MRP system. Paper orders are reproduced on the decorator HMI screen.
- Visualisation is limited.

One extra impact of the changeover of decoration is that the line has to be meticulously checked to make sure none of the previous batch are stranded on the conveyors to prevent a customer being sent a wrongly decorated can. The process of checking the line involves two employees walking the line one after the other, a process taking around 2 hours, to ensure that the line is 100% clear. This means that although decorator changeover is accomplished in approximately 30 minutes, an extra 1.5 hours of production time is lost. All of the 59 beverage can factories have one or more lines that runs small batches of cans.

4.3.5 Smartline

Crown has developed its own system for collecting production line data, “Smartline”. It consists of three components, see Figure 47:

1. An Open Platform Communications (OPC) server monitoring the PLC status information and logging it to produce time based information.
2. A database, data is stored in two ways for later reporting:
    a. Machine state (Running, Starved, Blocked, failed and changeover) is aggregated into 15 minute blocks.
    b. Failure data is stored as individual events.
3. Web Server which has two functions:
    a. It generates a “line mimic” displaying the current machine state and rate of production of the production line, served via HTML for devices connected to the Crown wide area network.
    b. Generates reports for historic production for analysis.

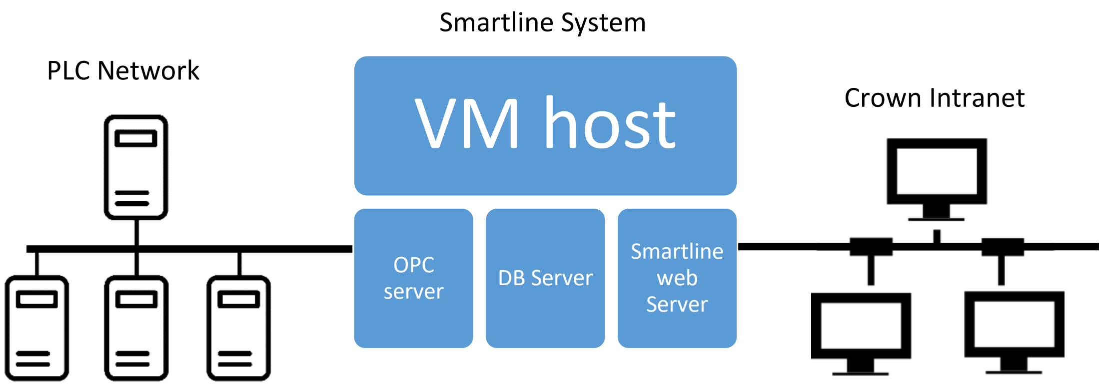


Figure 47 Crown Smartline

The line mimic is a key piece of technology and is displayed at many locations along the production line, giving staff an overview of the production line, see Figure 48.

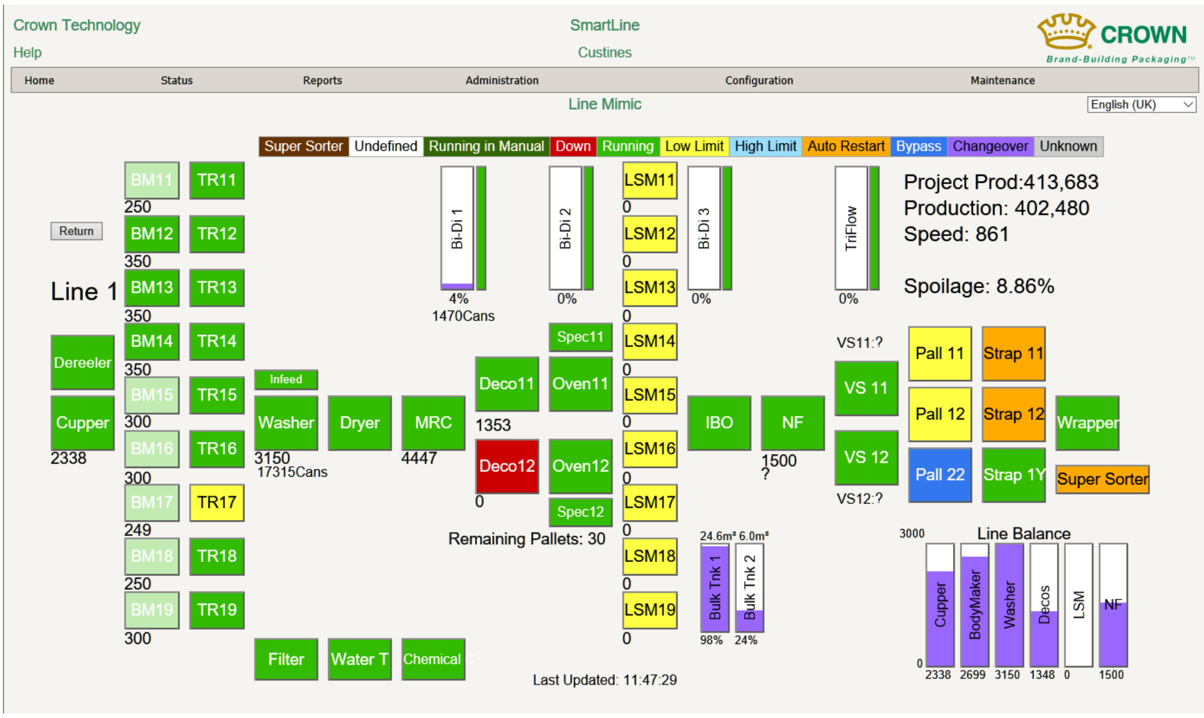
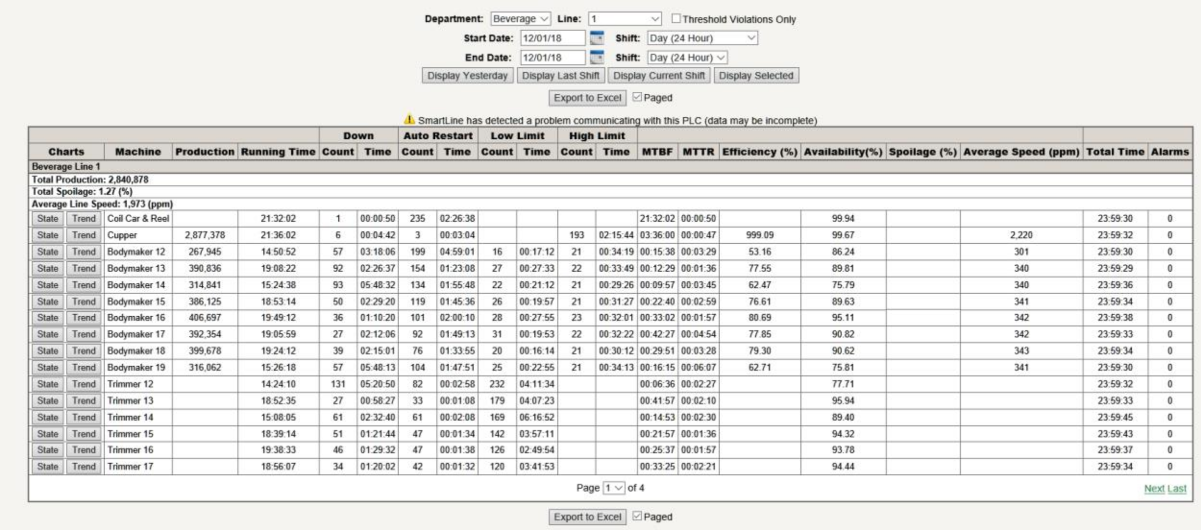


Figure 48 Line Mimic of Production Line

The Smartline system can generate reports and graphs of historic performance, see Figure 49 and Figure 50



Department: Beverage Line: 1 Threshold Violations Only
Start Date: 12/01/18 Shift: Day (24 Hour)
End Date: 12/01/18 Shift: Day (24 Hour)
Display Yesterday | Display Last Shift | Display Current Shift | Display Selected
Export to Excel Paged

SmartLine has detected a problem communicating with this PLC (data may be incomplete)

				Down		Auto Restart		Low Limit		High Limit									
Charts	Machine	Production	Running Time	Count	Time	Count	Time	Count	Time	Count	Time	MTBF	MTTR	Efficiency (%)	Availability(%)	Spoilage (%)	Average Speed (ppm)	Total Time	Alarms
Beverage Line 1																			
Total Production: 2,840,878																			
Total Spoilage: 1.27 (%)																			
Average Line Speed: 1,973 (ppm)																			
State Trend	Coil Car & Reel		21:32:02	1	00:00:50	235	02:26:38					21:32:02	00:00:50		99.94			23:59:30	0
State Trend	Cupper	2,877,378	21:36:02	6	00:04:42	3	00:03:04			193	02:15:44	03:36:00	00:00:47	999.09	99.67		2,220	23:59:32	0
State Trend	Bodymaker 12	267,945	14:50:52	57	03:18:06	199	04:59:01	16	00:17:12	21	00:34:19	00:15:38	00:03:29	53.16	86.24		301	23:59:30	0
State Trend	Bodymaker 13	390,836	19:08:22	92	02:26:37	154	01:23:08	27	00:27:33	22	00:33:49	00:12:29	00:01:36	77.55	89.81		340	23:59:29	0
State Trend	Bodymaker 14	314,841	15:24:38	93	05:48:32	134	01:55:48	22	00:21:12	21	00:29:26	00:09:57	00:03:45	62.47	75.79		340	23:59:36	0
State Trend	Bodymaker 15	386,125	18:53:14	50	02:29:20	119	01:45:36	26	00:19:57	21	00:31:27	00:22:40	00:02:59	76.61	89.63		341	23:59:34	0
State Trend	Bodymaker 16	406,697	19:49:12	36	01:10:20	101	02:00:10	28	00:27:55	23	00:32:01	00:33:02	00:01:57	80.69	95.11		342	23:59:38	0
State Trend	Bodymaker 17	392,354	19:05:59	27	02:12:06	92	01:49:13	31	00:19:53	22	00:32:22	00:42:27	00:04:54	77.85	90.82		342	23:59:33	0
State Trend	Bodymaker 18	399,678	19:24:12	39	02:15:01	76	01:33:55	20	00:16:14	21	00:30:12	00:29:51	00:03:28	79.30	90.62		343	23:59:34	0
State Trend	Bodymaker 19	316,062	15:26:18	57	05:48:13	104	01:47:51	25	00:22:55	21	00:34:13	00:16:15	00:06:07	62.71	75.81		341	23:59:30	0
State Trend	Trimmer 12		14:24:10	131	05:20:50	82	00:02:58	232	04:11:34			00:06:36	00:02:27		77.71			23:59:32	0
State Trend	Trimmer 13		18:52:35	27	00:58:27	33	00:01:08	179	04:07:23			00:41:57	00:02:10		95.94			23:59:33	0
State Trend	Trimmer 14		15:08:05	61	02:32:40	61	00:02:08	169	06:16:52			00:14:53	00:02:30		89.40			23:59:45	0
State Trend	Trimmer 15		18:39:14	51	01:21:44	47	00:01:34	142	03:57:11			00:21:57	00:01:36		94.32			23:59:43	0
State Trend	Trimmer 16		19:38:33	46	01:29:32	47	00:01:38	126	02:49:54			00:25:37	00:01:57		93.78			23:59:37	0
State Trend	Trimmer 17		18:56:07	34	01:20:02	42	00:01:32	120	03:41:53			00:33:25	00:02:21		94.44			23:59:34	0

Page 1 of 4 Next Last

Export to Excel Paged

Figure 49 Report of Historical Performance generated by Smartline

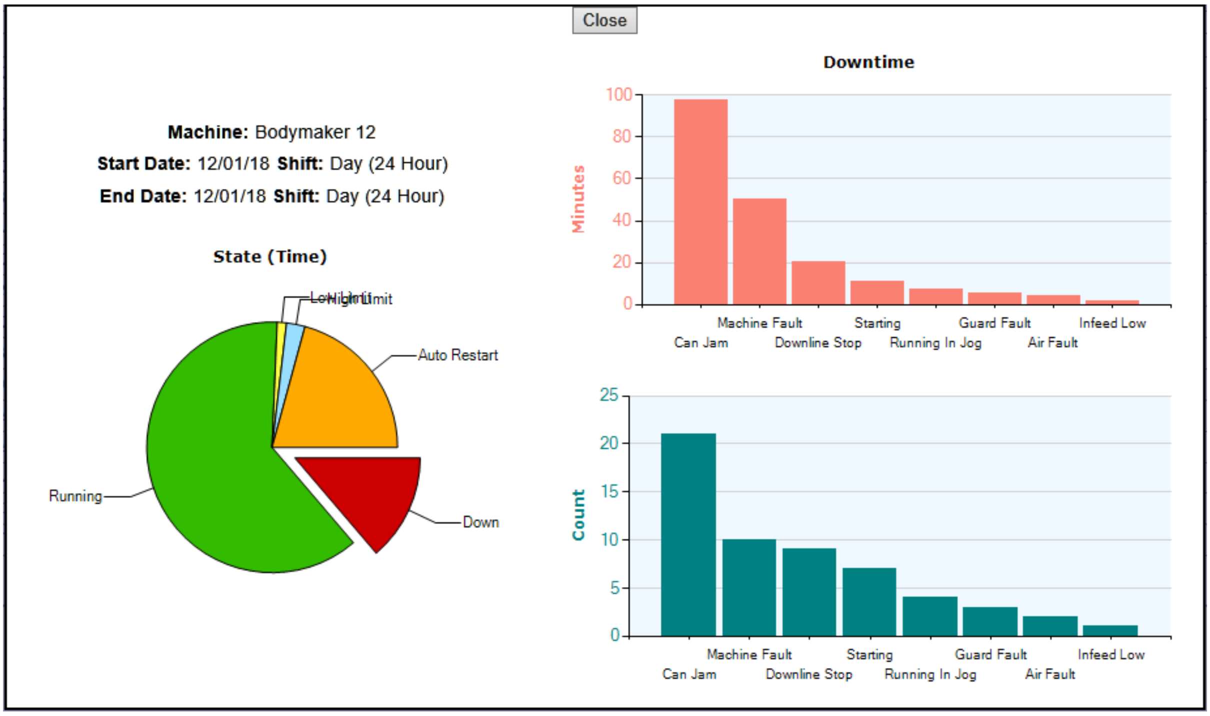


Figure 50 Chart of Machine State generated by Smartline

As it is based on PLC collected data, whilst this makes the data reliable, it has a fairly low collection frequency which would make it unsuitable for certain types of diagnostic sensor.

The system is a monitoring system only and no messages travel down from the high level system into OPC network.

5 Analysis of Results

5.1 Objective 1

Review of state of the art case examples pertaining to digital twin for high speed, almost continuous, manufacturing systems, particularly within the food and drinks and packaging sector.

This objective was investigated in the literature review and interviews with Crown experts. The literature review showed that there were many case studies carried out on the feasibility of creating a digital twin in terms of a "true digital twin" (Lee, Bagheri and Kao 2015) with real time data links to the physical system (Vachalek et al. 2017; Yin, Stecke and Li 2018; Zhuang, Liu and Xiong 2018).

The interviews with Crown experts and insight gained from empirical experimental work show that Discrete Event Simulation can be driven from several SQL data sources in order to create an automated data link from the physical system and the virtual representation.

There is a possible use case for the Discrete Event Simulation based closed-loop Digital Twin to increase productivity in that most beverage can factories have one line which produces relatively small batches of can, of the order 420,000 cans, which results in decorator changeover 1 or 2 times per shift. If the decorator changeover occurs at the same time as shift change, then the disruption can lead to increased downtime of the system of approximately 30 minutes. The cost in production for this extra time is approximately $675, based contribution of $15 per 1000 cans and decorator speed of 1500 cans per minute. Assuming this clash occurs once per day, this represents approximately 3% of annual profit contribution of $8million per line which equates to approximately $240,000 per annum per line. Over 65 production lines, this is approximately $15.6million per annum globally in lost production for Crown.

The system could optimise the order of production so as to reduce the number of changeovers that will be affected by shift changes. However is was found that proceeding to develop a Discrete Event Simulation closed-loop Digital Twin would be less effective than developing the existing Smartline system because stochastic based analysis is not useful for near continuous production systems without the benefit of flexible manufacturing routes.

5.2 Objective 2

To develop a baseline simulation model to investigate opportunities for theoretical improvement in the production systems, in order to create a quick win and obtain company-wide buy in and as a manual platform to develop an understanding of the requirements for a Digital Twin.

This objective was met in the Discrete Event Simulation experimentation, literature review, and interviews with Crown experts.

Both Discrete Event Simulation projects at Braunstone and Custines, demonstrated that the use of baseline simulation model identified opportunities for theoretical improvement in the production systems.

The Braunstone and Custines experiments demonstrated the use of Robinson method (Robinson 2011) in designing and managing a robust Discrete Event Simulation experiment, see Figure 51

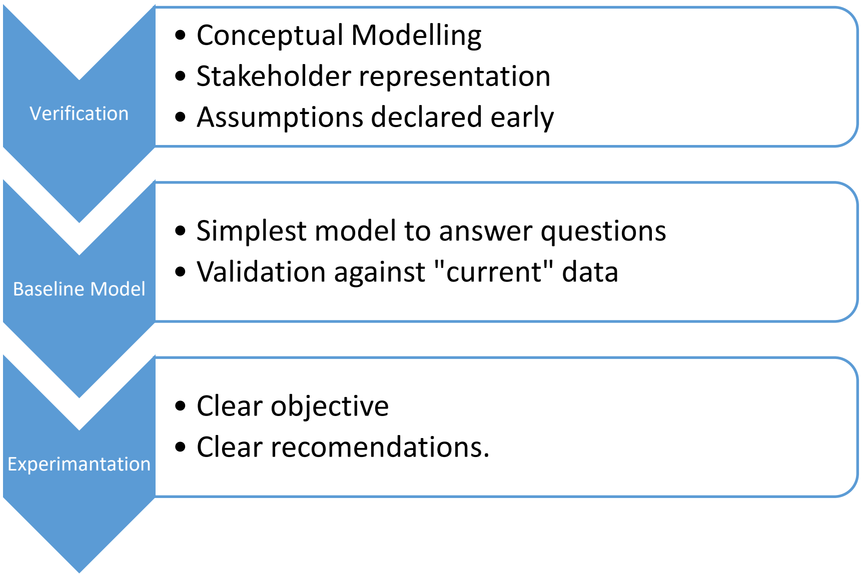


Figure 51 Robinson Method for Discrete Event Simulation

The capital expenditure experiments, where it was demonstrated that 9 Jefferson machines would be required to maintain production rates at Braunstone and that the optimum number of Bodymaker machines is 9 at Custines, indicated clear recommendations whether positive or negative in outlook, thus enabling statistical based analysis to support Crown management decision making process.

The Custines model showed that it could be driven from input data from a spreadsheet in units familiar to Crown stakeholders, e.g. cans per minute instead of a cycle time, and therefore the model could be disseminated for use by staff without extensive Simio training if they change the input values within Excel. The main KPI, Throughput, and other outputs, for example charts of machine utilisation, are generated in Excel after the output from Simio is post-processed by a Perl script. Whilst this can be initially tedious, this process could be automated by creating a more sophisticated Perl program to parse through the output file and create the report including charts directly including bar charts generated directly as demonstrated by the code by Bruening (Bruening 2017) .

The Custines work demonstrated an innovative methodology in the use of Discrete Event Simulation at Crown:

- Use of an input data file to reduce the hard-coding in the Discrete Event Simulation model. The data file contained line specific information, for example conveyor lengths and speed which would allow for easy verification by line experts not familiar with Discrete Event Simulation code.
- Replication of line control logic. The control of the line was a subroutine event triggered every 1 second to adjust speed of machines and check levels of BiDis and conveyors.
    - The decorator group speed was matched to the throughput of the Bodymaker group.
    - The Necker machine was matched to the throughput of the LSM group.
    - The levels of the BiDi were regulated to allow empty out of conveyor ovens.
- A method to pre-populate the model based on a data source, which could include the state, in terms of work in progress, of the physical production line.

In this way it demonstrates that the Discrete Event Simulation code Simio can be utilised as a Digital Twin:

- It can read from model and experimental data from a tabulated data source, either file or SQL.
- It can perform experimentation with the required replications.
- It can automatically generate a report enabling statistical based analysis to support Crown management decision making process.

5.3 Objective 3

Appraise the production line control systems for Crown's production facilities that the baseline simulation model is reflecting.

This was examined through discussions with Crown experts in this area and by experimentation with the Custines model.

The process line controls are implemented to protect can damage in three ways:

1. Stopping machines where the output conveyor is already full.
2. Stopping entry into conveyor ovens when there is no escape capacity remaining.

3. Modulating machine speed to reduce stop/start behaviour.

The baseline model was compared to a model with no controls, which represents the theoretical maximum output of the line. The uncontrolled model has an output only 6.7% greater than the controlled baseline model. This suggests a minor possible impact to improving the control algorithm with a high risk of damage to cans.

A further experiment examining the effect of utilising more of BiDi 3 capacity for the purpose of buffering failed to show any increase in throughput despite the extra spoilage.

5.4 Objective 4

Establish whether the information is available from the control systems to supply input data to a digital twin simulation model.

This objective was examined in the discussions with Crown experts.

The Smartline process monitoring system stores information in a Microsoft SQL database which can be read by Simio directly. The static information, such as conveyor lengths could be read directly from Excel data sheets.

The digital visualisation of the physical production line, in Crown parlance “Line Mimic”, can already be considered a Digital Twin(Aitken 2018), albeit of type 3 characteristics: with only visualisation and data storage capabilities.

A simulation based Digital Twin in order to optimise the timing of the sales orders being produced would require information from four sources:

1. Smartline. The system has the failure data and the current state WIP of the production line. It can be accessed via SQL and could populate the model and set failure model parameters.
2. Line Controls. This is the PLC program which holds the control logic for the production line. It is fairly static information which would require manual interpretation for the Digital Twin.
3. Physical Layout. Depending on the age of the facility, this is in in the form of either CAD or block diagrams. This information is static and would require manual interpretation for the Digital Twin.
4. Financial Control System. This is a network based system, either JD Edwards or SAP, which can be accessed via SQL through an Application Programming Interface (API).

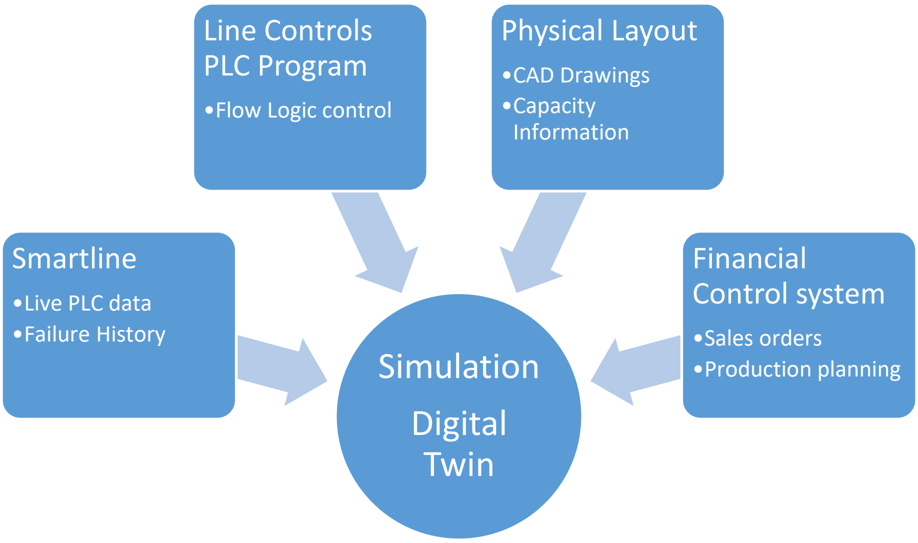


Figure 52 Crown Digital Twin Data Sources

If the Digital Twin was required to adjust Line Control parameters and execution speed of the simulation became an issue because the decision time of the system was less than the simulation run time, see Table 6 Scale Experiment Results, it could be possible to run the simulation in a number of scenarios to generate a decision matrix or response surface. The response surface would be used to drive the system in real time as demonstrated in the feasibility articles (Rivera-Gómez, Gharbi and Kenné 2013; Sajadi, Esfahani and Sörensen 2011).

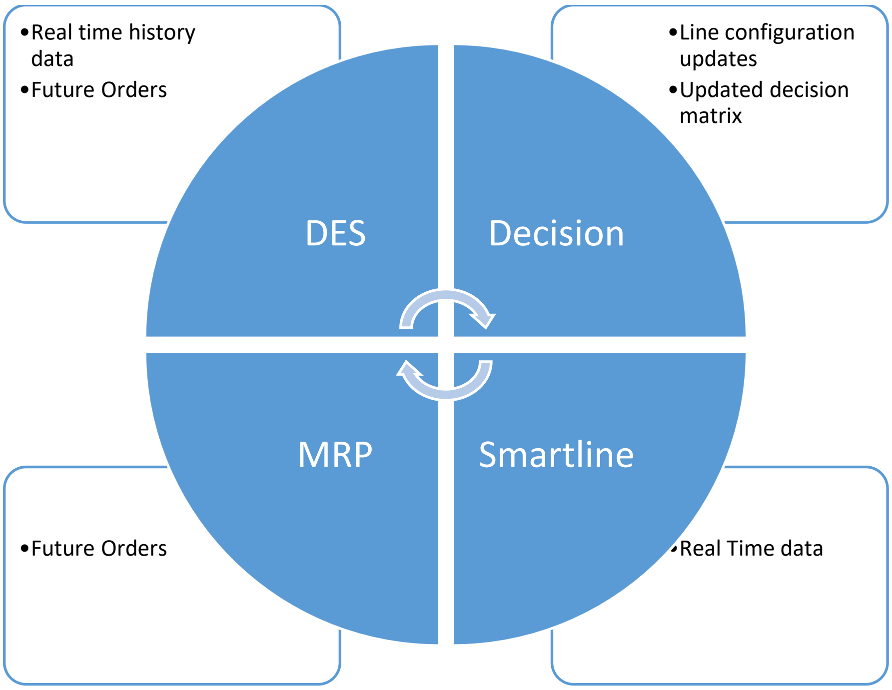


Figure 53 Smartline with Autonomous Control

Crown could enhance the Smartline system at relatively little cost:

I. Include more of the physical architecture of the production line, for example conveyor sizes and loading. The extra information pertaining to work-in-progress would be beneficial.
II. Move the order control system from the PLC architecture in to Smartline. This will change Smartline from being simply a passive monitoring system, to a control system. The enhanced visuals and ability to monitor from anywhere in the world would be a major benefit over the PLC system.

These extra pieces of information could be utilised to pre-populate a Discrete Event Simulation model, almost directly, in order to run the Digital Twin for order planning, particularly on lines with smaller batch sizes for optimising change-over sequences.

5.5 Objective 5

Identify what other Industry 4.0 techniques and technologies would help Crown to reduce waste and increase throughput.

This objective was examined in the literature review and in discussion with Crown experts. There were three main areas which would be of benefit to Crown:

1. Smart Conveyors
2. Edge Computing
3. Big Data Analysis

5.5.1 Smart Conveyors

In terms of intelligent conveyors, as the Crown manufacturing process is linear with fixed process order, there seems little scope for implementation. There is however one case for a smart conveying system where the line is producing relatively small batches of cans of the order of 420,000 cans. The decorator change-over takes approximately 30 minutes but as the conveyors empty, some cans become stranded on the strips between belts. The process of changeover requires the line to be emptied out in order to prevent cans from the previous batch being packed on to the next customer pallet. This extra work, a process which takes approximately of 2 hours as the line has to be checked for stranded cans, means the decorator is idle for around 1.5 hours per changeover.

The ability to intelligently separate batches on the conveying system, see Figure 54, would allow the decorator to start the next batch whilst the previous batch is finished and palletised and allow for an extra 1.5 hours of production per changeover.

In order to accomplish batch separation, the conveying system would need:

- To ensure that cans could not stall between the conveyor sections as currently happens on the conveyors.
- Each section would have to sense when it was empty of cans.
- The conveyors would have to detect, transport and dispose of fallen cans.
- Intelligently track and maintain separation of the batches as the batches moved down the production line.
- Store historic can quantity and conveyor speed data for analysis.

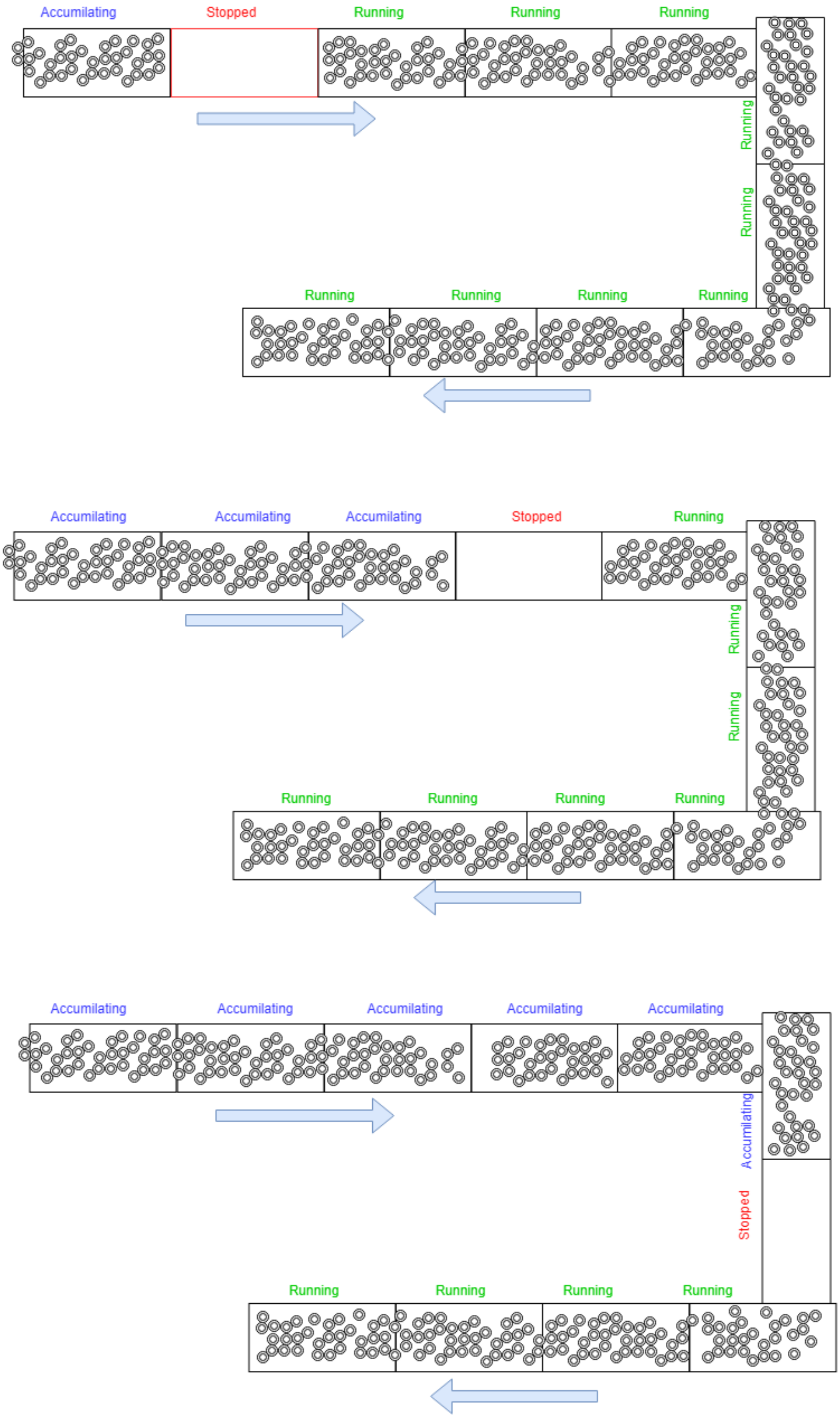


Figure 54 Smart Conveyor keeping batches separated

Crown has already developed its own system to monitor the status of machines on a production line, Smartline. Missing from the system is visualisation and history of the work in progress (WIP). The WIP is a considerable number of cans, averaging 70,000 and peaking at 104,000 cans, on Custines line 1. Whilst this may not represent a huge cost, if the numbers are constantly high, it could reflect a problem with the line controls system. The extra data from a smart conveying system showing the WIP on the conveyors would give a useful insight in to the state of the system as a whole and possibly lead to further optimisation.

5.5.2 Edge Computing

The literature review showed examples of edge computing benefits:

- Filling in the data collection gaps where legacy equipment is not OPC compliant.
- Filling in network gaps where the cost of installing network ports to legacy equipment is expensive. This could also employ node-to-node communication, where the wireless communication was inadequate.
- Higher frequency data collection where the amount of data would overwhelm the network bandwidth. An edge computer could have storage appropriate to the volume of information required. The data could be analysed at the edge, compressed for transmission to the cloud system, or physically collected for analysis.

In terms of Crown's production, the beverage can making facilities have a high degree of automation, as necessitated for the manufacturing speeds required to remain competitive. There are very sophisticated PLC control systems which are networked and able to communicate up and down the production line because they have standardised on a single PLC manufacturer. They have introduced an internally developed data capture and analytical system and are at the early stages of "big data" analysis. They are beyond Industry 3.0 and are tentatively in the Industry 4.0 stage, certainly at the visualisation level. The food and speciality packaging businesses remain firmly at Industry 3.0. Although they have a high level of automation, unlike the beverage facilities, they have limited, and non-standardised, PLC controllers. There is a very little networking of PLCs due to non-standardisation of PLC manufacturers. Additionally, some factories require extensive upgrades to the Ethernet network on the shop floor, which has proved to be a prohibitive cost and has in most cases prevented them from implementing Crown's Smartline system.

Edge computing via wireless communications would allow Crown to collect data from a variety of legacy systems.

5.5.3 Big Data

Crown has many factories with similar layout and very similar equipment with some examples being legacy equipment, for example it operates approximately 500 Bodymakers.

There exists an opportunity to experiment and possibly find a closed loop control system to improve productivity which could be scaled up across sites around the globe.

Retrofitting instrumentation to a group of Bodymakers, capturing data at a suitable frequency, which may require edge processing as it may be higher, depending upon the age of the PLC, than the PLC scan rates. With large amounts of data, it may be necessary to store locally and analyse later, or to use to use the edge computer to post-process the raw data to a smaller data file to upload to the cloud (McDonnel 2013).

If this example of Big Data experimentation fails to elicit a clear response, it may be possible to increase the data set further to include more groups of Bodymakers, employing the clustering and fuzzy clustering algorithms to account for the various age and manufacturer of the machines (Sassi Hidri, Zoghlami and Ben Ayed 2018).

With patterns of data and responses emerging, it may be possible to progress to develop an overview and edge based closed-loop monitoring system based on the large and continuous flow of data into the cloud based system, see Figure 55.

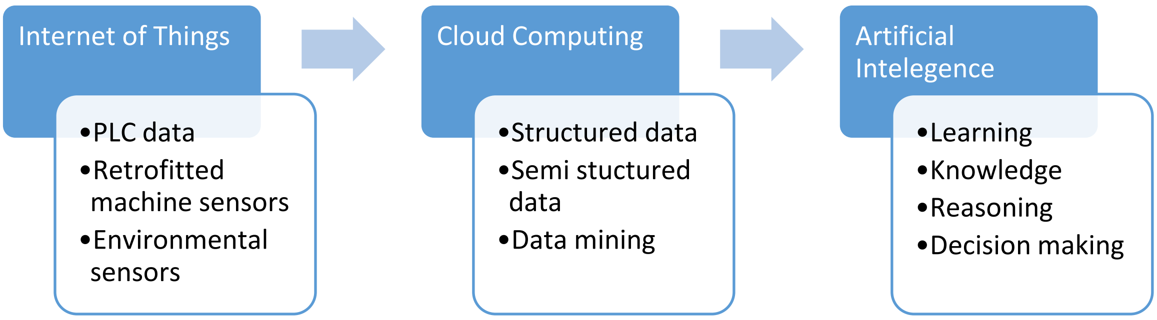


Figure 55 from Retrofitting Sensors to Artificial Intelligence

6 Discussion

6.1 Objective 1

Review of state of the art case examples pertaining to digital twin for high speed, almost continuous, manufacturing systems, particularly within the food and drinks and packaging sector.

This objective was investigated in the literature review and interview with Crown production line experts. The concept of the Digital Twin is, at the time of writing, a new concept. In a review of papers most Digital twins are categorised as "concept" studies (Kritzinger et al. 2018).

The research showed that there are very few actual examples of autonomous closed-loop Digital Twins in high speed manufacturing systems. Most examples, mostly theoretical, are based on a flexible manufacturing scenario where the artificial intelligence implements an alternative manufacturing route to keep the system running (Rosen et al. 2015). One possible reason for this would be the fixed process route to manufacture does not give an opportunity to for any intelligent system, human or artificial, to make a difference.

There are relatively few examples of a Digital Twin with full autonomy. Most (Kritzinger et al. 2018; Rosen et al. 2015) are theoretical concepts on what could be achieved, irrespective of cost, with a fully flexible manufacturing system. The Discrete Event Simulation software developers (Austin 2019; Simio LLC 2017a; Simul8 2018) envisage a system where the user is required to run the experiments and then feed the results back to the control system. In this way users can optimise the order book in terms of the pre-programmed logic of the simulation code, which is reflective of the physical system.

Crown has already made steps towards a visualisation Digital Twin with the development of its proprietary production line monitoring software, Smartline. The system collects data from production line PLCs and displays it, via HTML, in the form of a line mimic, which shows the order and status of machines on the production line. More detail can be obtained on individual machines, such as recent production speed and reliability, by selecting a machine within the HTML environment. All data collected for all machines connected to the system is stored chronologically for later possible analysis. The system is under continuous development, and additional sensors, not necessarily associated with a specific machine, such as ambient temperature and humidity, are being added to the data capture capability.

The research has shown that there is a case for implementing this kind of Digital Twin with a view to optimising the batches of work so as not to conflict with shift change overs with an approximate value of $15.5million worldwide. It has also demonstrated an innovative method to prepopulate the model conveyors using up to date data from the physical system in order to be more representative of the physical system.

6.2 Objective 2

To develop a baseline simulation model to investigate opportunities for theoretical improvement in the production systems, in order to create a quick win and obtain company-wide buy in and as a manual platform to develop an understanding of the requirements for a Digital Twin.

This objective was investigated by use of Discrete Event Simulation case studies of two of Crown's production lines.

This work has demonstrated the use of Discrete Event Simulation and use of Robison method (Robinson 2014) to create experiments that will give Crown clear recommendations, as detailed in the Results section. This in itself is not novel but is accepted good practice. Crown has used Discrete Event Simulation modelling in the past and this work has made innovations in the use of Discrete Event Simulation for Crown in the use of data-file inputs entry for model parameters, and experimental parameters with two objectives:

1. A more generic model allows for a speed up the experimental process. Factories with the same topology in terms of the order of machines and buffers can utilise the same Discrete Event Simulation model with data values for each line in the data file.
2. The real data is not hardcoded into the Discrete Event Simulation model and therefore allows for non-software experts to examine and experiment with the data values using Excel to interface with Discrete Event Simulation, which is the Crown standard office software.
3. The data is portable to other suitably configured Discrete Event Simulation codes, which means that Crown is not locked into a single software source therefore introducing flexibility of software use for the organisation as a whole.

The innovative use of a data file means that the software can be used as the analytical device in a Digital Twin rather than be developed to become the Digital Twin. The Discrete Event Simulation software Simio can be run via a programming API, which allows for a pre-programmed experiment to be run (Simio LLC 2017b). The experiment can take input parameters from the data file and perform the necessary replicated runs. This would allow for the Discrete Event Simulation software to be scheduled or run autonomously within the control of other software for example Smartline, thus utilising the benefits of Discrete Event Simulation.

6.3 Objective 3

Appraise the production line control systems for Crown's production facilities that the baseline simulation model is reflecting.

This objective was investigated by the Custines Discrete Event Simulation experiment in which the baseline model was compared to a model with no controls. The results showed that the controls inhibit production by only 6.7%.

Adjusting the control of Bidi3 in order to use more of it as a buffer, failed to elicit the desired increase in throughput. This could be due to the fact that the extra buffering really has little or no effect on throughput, or there is a compensating control system which negates the effect of the change.

The line controls are primarily in place to reduce spoilage by reducing stop start behaviour and ensuring conveyors do not exceed maximum or minimum capacities. I have demonstrated that it is possible for the Discrete Event Simulation to model the actions of the line control system, but the link to the PLC based Line Control is the most challenging. Any improvements in control logic would have to be reinterpreted by a line electrical engineer for implementation on the PLC based system.

Crown management often bring up the subject of optimising line controls as part of a Discrete Event Simulation project, which is for two reasons:

1. The opaque nature of the PLC based system. Only a few individuals which set up a given system are able to verify its logic.
2. Experimentation can involve monitoring the production line for many hours. For this reason, the controls are often set conservatively and rarely amended.

It is theorised that future Discrete Event Simulation studies of Crown's production lines will lead to an improvement in certain production line controls and thereby in increase in throughput of those lines.

6.4 Objective 4

Establish whether the information is available from the control systems to supply input data to a digital twin simulation model.

This objective was investigated in discussion with Crown experts. The information required to build a Digital Twin is stored in four places, two of them are fairly static and the other are database applications.

- Smartline: A database application with live data of machine state available.

- Financial Control System: SAP or JD Edwards, both systems have a web API for obtaining sales order data.
- Physical Layout: CAD, Scanned drawing or block diagram. This data is difficult to obtain but static.
- Line Control Logic: This is stored as a PLC program but is often only understandable by relatively few experts.

The two static pieces of data, layout and control logic, could be collected on an ad hoc basis as a simulation project is identified, but this would be a mistake as part of the benefit of the Digital Twin is that it closely represents the physical system and therefore has more than a single use. In the example of the Crossrail Digital Twin, it was conceived for the purpose of viability and promoting funding; it was elaborated to simulate the construction; and it is planned to be used for operational purposes (Hibbert 2014). At each stage more data is added to the Digital Twin; data updated but not deleted; and it is available for more than a single purpose, and probably for more uses than the designers of the Digital Twin envisaged.

The experimental programming work, see section 6.2, has demonstrated that the use of Discrete Event Simulation software could be used to perform experiments generated by an autonomous Digital Twin.

6.5 Objective 5

Identify what other Industry 4.0 techniques and technologies would help Crown to reduce waste and increase throughput.

6.5.1 Smart Conveyors

The research identified the Crown requirement to keep batches separate from each other on the conveying system following decoration change. The requirements which would have to be met to allow batch separation, the conveying system would need:

- Not necessary to be bi-directional.
- To ensure that cans could not stall between the conveyor sections as currently happens on the conveyors.
- Each section would have to sense when it was empty of cans.
- The conveyors would have to detect, transport and dispose of fallen cans.
- Intelligently track and maintain separation of the batches as the batches moved down the production line.
- Store historic can quantity and conveyor speed data for analysis.

Crown places a lot of importance on conveying as effective conveying minimises can damage and ensures machines on the production line are, as much as possible, constantly supplied with work. Crown could use the bullet points above in negotiating with conveyor vendors or as a further research objective.

6.5.2 Edge Computing

Edge computing is defined as technologies that allow computation to be performed near the data source (W. Shi and S. Dustdar 2016).

The research has suggested two use cases for Edge computing:

Firstly if Crown wished to retrofit additional sensors to a legacy machinery to look for patterns of machine state and reject instances or failure, then doing so via an edge computing device (H. Derhamy et al. 2018) would allow certain advantages:

- Quicker implementation without the need to modify an existing PLC controller.
- Wi-Fi connectivity would mean reduced infrastructure cost to implement. Wireless, for example Bluetooth (Nilsson 2013), connectivity to a remote sensor further increases the flexibility of the Edge Computer.
- Data could be captured at a higher frequency and analysed at the edge (Trinks and Felden 2018), compressed for transmission to the cloud, or simply stored for later analysis. All aimed at reducing network traffic.

If no useful loop is detected, the Edge computer and sensors could be easily dismantled and moved to the next candidate for experimentation.

Secondly, as many factories are prevented from implementing any kind oy Industry 4.0 technology due the costs of computer network upgrades and connectivity to legacy industrial machines, edge computing could be used to acquire the PLC data and send it wirelessly to the cloud based Digital Twin of the facility, Smartline in the case of Crown. As the data is not being used in the critical control context, the extra latency of a Wi-Fi connection, compared to real time Ethernet, could be tolerated:

- Edge computer could compress data and send summary to Smartline system to unpack
- Edge computing could extend network if WIFI blind spots are identified as a problem, termed as “clustering” (Alnoman et al. 2018), where a cluster head node manages communications of wirelessly connected sub-nodes.

Crown has already taken the decision to develop its own monitoring software, Smartline, and it could look into developing its own edge based systems based on single board computers to capture data from legacy equipment. With about 55 food, aerosol, and speciality packaging factories worldwide, the cost saving from licensing could offset the cost of development.

6.5.3 Big Data

Big data is defined as data source, or multiple data sources, which have extremely large volumes of data, extremely high velocity of data, and extremely wide variety of data (Hurwitz 2013).

As alluded to in section 3.2.2 Crown manufacturing efficiency is based upon constant machine availability. In order to improve efficiency, Crown could look in to two areas:

1. Work more openly with suppliers to find possible external causes for high spoilage.
2. Smart sensors and data analytics targeted at finding links between specific machine states and high spoilage.

To put the first point in to context for Crown, high spoilage in the Bodymaker process, could be related to variations in material properties when the coil of aluminium is manufactured. By sharing data, it could be possible to manufacture in a way that improves the Crown production process, or alternatively, optimise the Crown process to suit the manufacture of the coil.

An example of the second point would be to add extra sensors to a suspect process, and processes upstream to investigate possible causal links which are at the moment not identifiable. The data set could be widened by the use of clustering (Lapira 2012) and the technique of fuzzy clustering (Frieß et al. 2018) allows for non-identical machines to be compared. For example Crown employs Bodymaker machines of different age and manufacturer, which could be clustered. Fuzzy clustering would group the machines which behave in the same manner in order to compare performance.

7 Recommendations

7.1 Smartline Response Time Capture

Smartline captures fault information in order to calculate mean time to failure and mean time to repair as discrete database entries. During a typical fault event, for example when a Bodymaker detects a tear-off, the PLC will report a number of changes in state as the machine is isolated, guards opened, until the machine is put back into a running state. Smartline captures the initial cause, i.e. the tear-off fault, and the total duration of the fault to store in the database, see Figure 56.

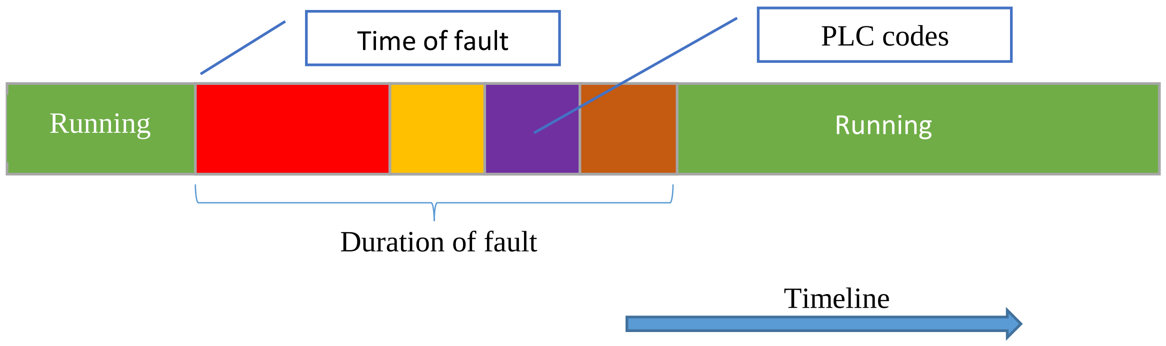


Figure 56 Smartline Current Error Aggregation

Aggregating the faults saves storage space in the database, as most of the PLC codes are not relevant. However, a crucial bit of information is lost. During the fault event there will be a specific point in time when the machine is first isolated or a guard opened. The time between the initial fault and this time of first intervention is the human response time, see Figure 57 Proposed Error Aggregation. The time from the initial response until the machine is back running is the actual repair time.

The response time could be valuable information with regard to manning levels on the line and could easily be stored in the database with only fractionally more storage requirement. The extra

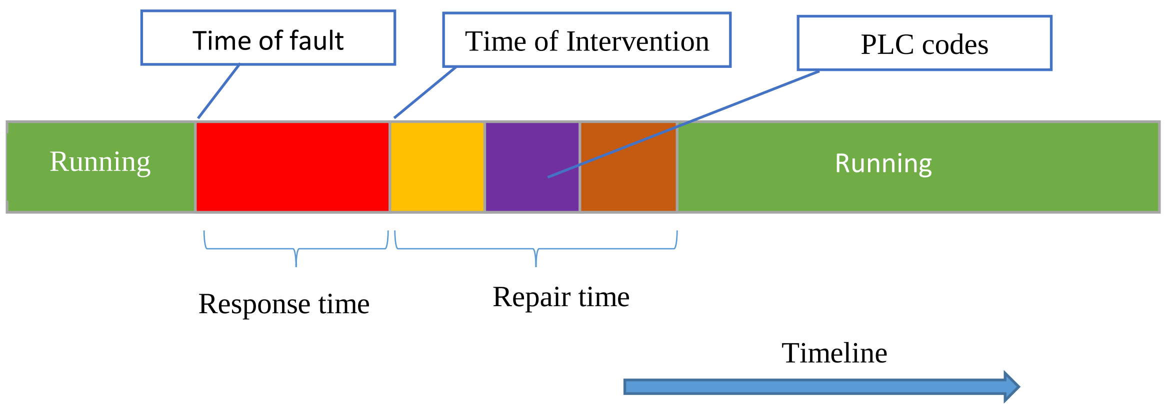


Figure 57 Proposed Error Aggregation

information could be used to include manning levels in the simulation. Although more logic and other data would be required, the system would quickly generate validation data for a new baseline model.

7.2 Smartline Partial Pallet

Crown has a pallet counting system implemented as a PLC program, in a single beverage factory at Botcherby. The system is used to stop the decorator when the work in progress is enough to complete an order and prevent over production.

It was a successful implementation and could be repeated at other Crown facilities but it has some minor drawbacks:

- Increased licensing cost for extra PLC input output screens (HMI).
- Not directly repeatable. Due to the diverse nature of PLC program structure.
- Paper system. There is no direct link to MRP system. Paper orders are reproduced on the decorator HMI screen.
- Visualisation is limited.

If the system was moved from the PLC domain to the Smartline system. All of the above points could be addressed:

- HMI screens for decorator and palletiser would be web based with no license costs.
- The Smartline version would be available to implement to all factories with Smartline system.
- Data entry for orders would take place in the office environment, not on the shop floor. Additionally the entry form would be modular to allow links to the MRP system, if available.

Visualisation of conveyor capacity would be available to everyone. Via a web mimic, see Figure 58 Smartline Order control.

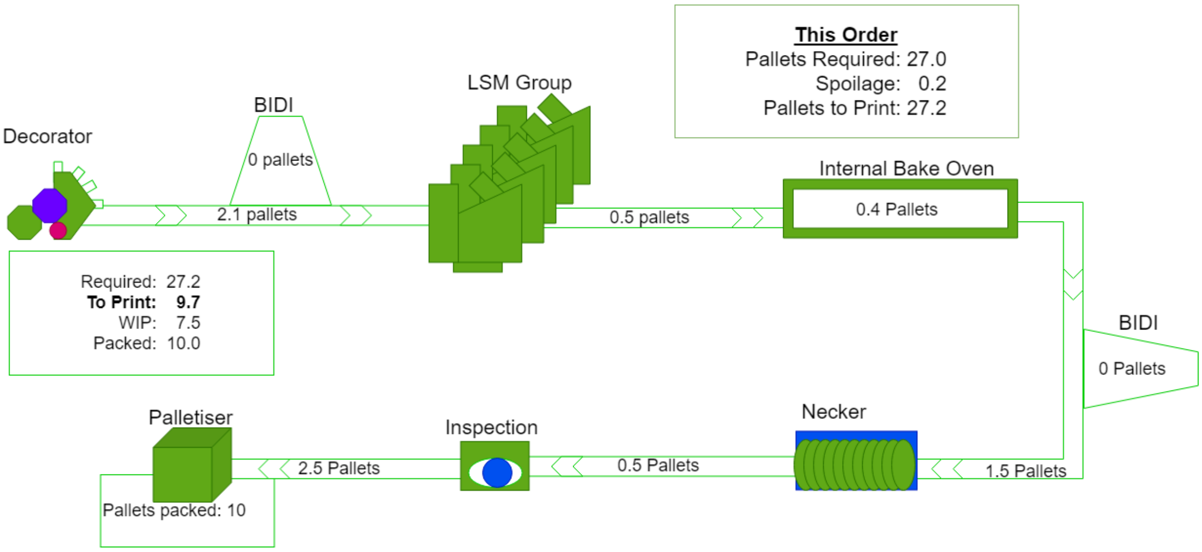


Figure 58 Smartline Order control

7.3 Smartline Future Enhancement

Crown has a proprietary machine monitoring system in Smartline which allows for real time monitoring and archive of machine state data for later analysis, all of which can be performed anywhere in the World via the web based interface.

The lack of conveyor length physical data and the fact that line control logic and control is contained in an opaque PLC based system, means that any experimentation with regard to the control system cannot be conducted at a higher level than the factory level.

Crown could enhance the Smartline in two ways in order to enhance it as a Digital Twin:

I. Include more of the physical architecture, for example conveyor sizes and loading. Crown has the data and simply needs to be convinced that it would be useful for operational purposes to include it in Smartline. The conveyor data should be displayed on Smartline and be available for analysis.
II. Move the conveyor control system from the PLC. With the logic more accessible, it could be compared across factories with the possibility of experimentation to determining a generic solution.
III. Move the order control system from the PLC architecture in to Smartline. This will change Smartline from being simply a passive monitoring system to a control system. The enhanced visuals and ability to monitor from anywhere in the world would be a major benefit over the PLC system.

These extra pieces of information could be utilised to pre-populate a Discrete Event Simulation model in order to further enhance the Digital Twin for order planning.

Additionally with more information in the same format, it should be relatively easy to implement machine clustering and optimisation from a global viewpoint. The higher level visualisation Digital Twin could be developed to collect data from a number of factories which would give an overview of a region rather than a single factory. An additional benefit would be the ability to visualise and compare lines based upon product which were not necessarily under the same factory roof.

The work to include a data file to drive the Discrete Event Simulation code could be taken further to include the line control logic in to the data file, so long as a generic representation can be found. In the same vein, the output from Simio should be programmed so as to simplify the report generation.

Crown could assimilate production line data etc., into a simulation software based Digital Twin in line with the view of the software vendors, but I think this would be a mistake. It would suffer from the same problem as the PLC Line Control system, in that it would be opaque to non-experts. Smartline, wherever possible, should be the central repository for data as it is web based and therefore accessible to anyone, given security access, on the Crown intranet. Crown has more than a hundred factories that could benefit and as Smartline does not attract license fees is a cost free scalable solution.

8 Conclusions

8.1 Objective 1

Review of state of the art case examples pertaining to digital twin for high speed, almost continuous, manufacturing systems, particularly within the food and drinks and packaging sector.

This objective was investigated in the literature review and interview with Crown production line experts.

The research showed that most examples of Digital Twins are aimed at optimising the routing of goods through the use of a flexible manufacturing system. This would suggest that the applicability of developing and implementing an autonomous Digital Twin at Crown would be limited as there is no scope to alter manufacturing flow based upon an event on their continuous large batch process lines.

The research did shown that there is a limited case for implementing a Digital Twin on process lines with small batch runs of approximately 400,00 cans, with a view to optimising the batches of work so as not to conflict with shift change overs, with an approximate value to Crown of $15.5million worldwide. It has also demonstrated this could be achieved with an innovative method to prepopulate the Digital Twin model conveyors using up to date data from the physical system in order to synchronise the model with the physical system.

8.2 Objective 2

To develop a baseline simulation model to investigate opportunities for theoretical improvement in the production systems, in order to create a quick win and obtain company-wide buy in and as a manual platform to develop an understanding of the requirements for a Digital Twin.

The use of Discrete Event Simulation to investigate opportunities for theoretical improvement in the production systems has been achieved. The study of the Braunstone factory showed the feasibility of adding a new process into the production line and quantified the number of stations required, given the assumptions on data.

The Study at Custines factory showed the viability of adding an extra Bodymaker to the production line, with a payback of investment of between 215 and 250 days, and disproved assumptions with regard to increasing buffer sizes.

The innovative use of the data file:

- Allows for the Discrete Event Simulation model to be more generic and reused for production lines with similar topology.
- Allows for stakeholders, not trained in Discrete Event Simulation, to examine more of the model data.

This could be further enhanced by moving Line Controls data into the input file, for example conveyor and Bidi threshold values.

The research demonstrated the innovative use of remote computing technology to give the benefits of a web based system, i.e. a system that is always on and can perform 20 concurrent replications, without the additional cost of simulation software licensing.

If Smartline was developed so as to hold the Line Control parameters, both the PLC system and the simulation system could be developed with a common data format, see Figure 59. This would allow for experimentation with reduced risk of transposition and logical errors as results are reinterpreted between systems. With Line control data stored in a single location as opposed to various PLC computers, it would make the goal of an autonomous Digital Twin including line control optimisation easier to accomplish.

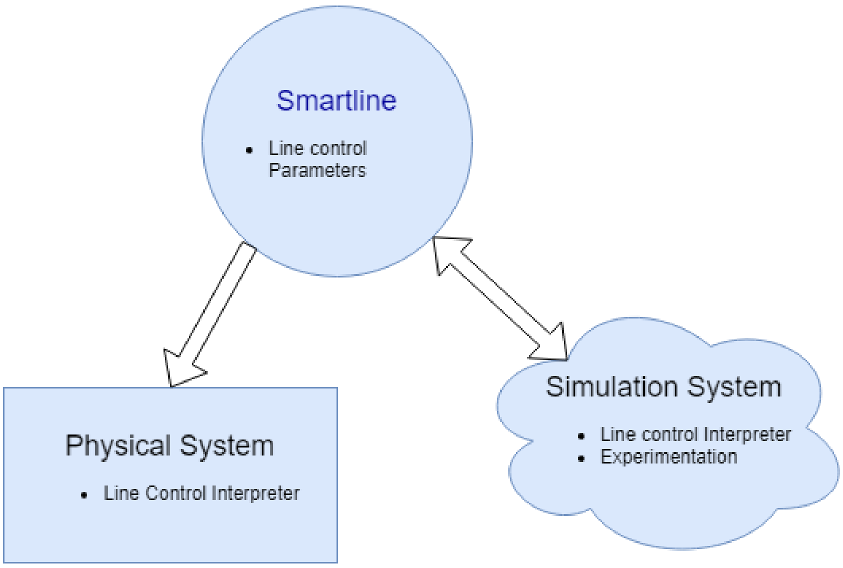


Figure 59 Physical and Virtual Systems with Common Data format

8.3 Objective 3

Appraise the production line control systems for Crown's production facilities that the baseline simulation model is reflecting.

The innovative method of programming the line control in the Discrete Event Simulation code worked well. It was demonstrated that line control parameters can be set up and mimic the physical system in order to validate the baseline model and perform experiments.

It was determined that Custines benefits from very effective line control logic which meets the requirement of minimising can spoilage and at the same time only inhibiting throughput by 6.7% compared to an uncontrolled system.

Further experimentation utilising more of BiDi 3 as buffer space failed to yield any extra throughput, but the model served as a useful tool for testing Line Control hypotheses.

When Custines was selected as a candidate for a line controls experiment, it was a new factory in the process of commissioning new equipment on the line. The lower than expected throughput was not due to poor line controls. With hindsight, Custines was not a good candidate for the study and it is suspected that there are better candidate factories for future work in this area.

On reflection, more work could have been put in to creating a generic control system and moving the input data, in terms of BiDi and conveyor threshold levels, to the input data file.

A lot of experimental time was spent creating a custom state variable to monitor the switched state of a machine – low limit, high limit, low speed, line speed, high speed – to more closely mimic the line control terminology, which turned out not to be as useful as envisaged.

8.4 Objective 4

Establish whether the information is available from the control systems to supply input data to a digital twin simulation model.

The research showed that the data is available to create an autonomous Digital Twin.

- Smartline: holds current machine state and historic failure data.
- Financial Control System: SAP or JD Edwards via a web API for obtaining sales order data.
- Physical Layout: CAD, Scanned drawing or block diagram.
- Line Control Logic: This is stored as a PLC program but is often only understandable by relatively few experts.

The fully autonomous Digital Twin is probably a few years in the future for Crown. It should concentrate on developing Smartline to encompass more to the physical system data, for example conveyor size and sensor placement; and Line controls logic, to allow it to evolve into a Supervisory Control and Data Acquisition (SCADA) system.

With these improvements it would be possible to include a simulation as an embedded subsystem to perform experiments to both optimise Line Controls and order scheduling, see Figure 60.

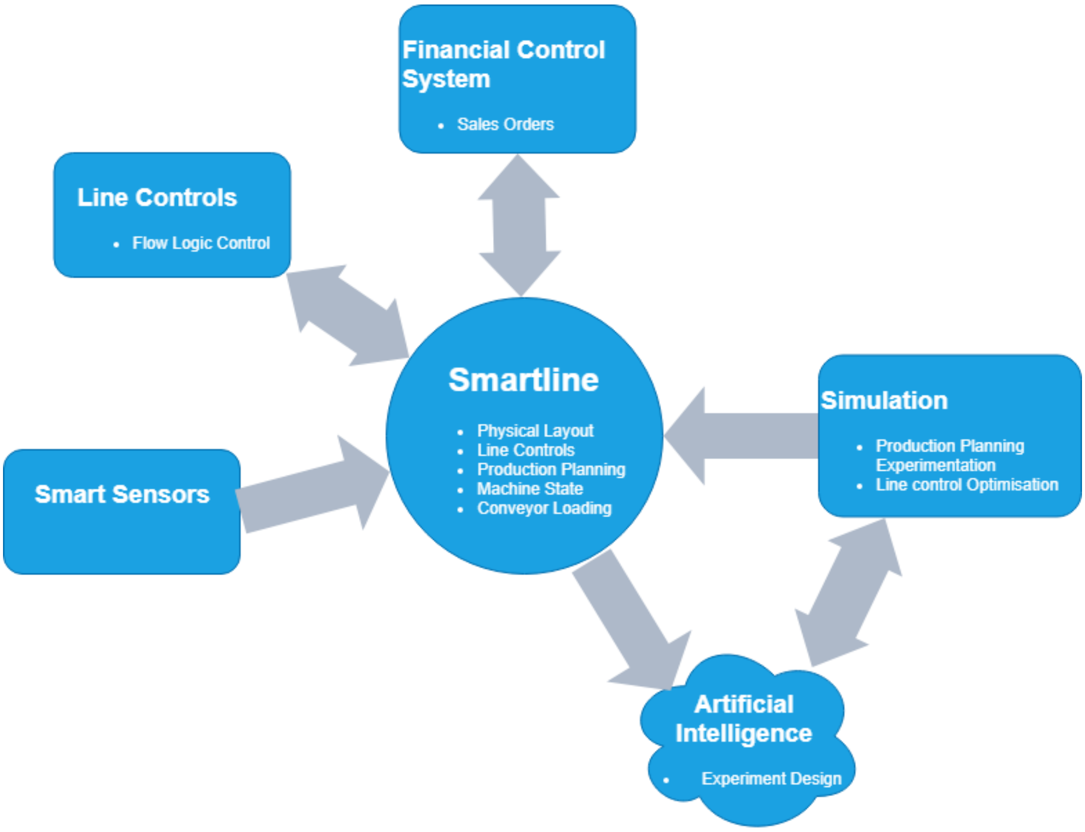


Figure 60 Smartline Trajectory

8.5 Objective 5

Identify what other Industry 4.0 techniques and technologies would help Crown to reduce waste and increase throughput.

This objective was accomplished in the main part by literature review and with interviews with Crown line experts giving context.

Industry 4.0 techniques are becoming ubiquitous. With cost of instrumentation and wireless connection driven down in the domestic personal computing market, the industrial applications, e.g. smart sensors are also benefitting from cost reduction, although there is still a premium for industrialised versions. I did identify Industry 4.0 techniques which could be explored further:

1. Improvements to Smartline so as to increase its usefulness as a Digital Twin.
2. Edge computing as a tool to deploy Smartline in factories with legacy equipment and limited network capacity.
3. Big data analysis by utilising data across many sites, including clustering of similar machines

9 Further Work

It is postulated that moving the Line Control logic to Smartline system would be beneficial. Future work in this area should determine feasibility, in particular:

- Reliability: The reliability of the software and network has not been an issue for Smartline as a monitoring only system. As a control system, it would need to maintain reliable contact with the PLC network. Any system should include a simple fall-back system, where the PLC reverts to a local control algorithm, if contact is lost with Smartline.
- Cost study: The value of having algorithms visible rather than hidden is sometimes a difficult cost to justify. The value of being able to experiment more easily and possibly find a generic solution to the line control algorithm may be found, should be taken into account.
- Autonomy: As Crown is the developer of Smartline, it could be possible to make future version with an artificial intelligence system embedded.

With so many factories with similar machines collecting data, a future big data project should be considered looking at the concept of clustering, and fuzzy clustering, to compare machines globally. It might be possible to identify factories with best practices and operating conditions in order to raise the output of other facilities.

10 List of Appendices

1. Ethical Approval
2. Braunstone Specification Document
3. Custines Specification Document
4. Large Server
5. Small Server
6. Batch/un-batch node
7. Crown Conveyor

11 References

Anon(2016) *Bentley Systems, Incorporated; Crossrail's Common Data Environment Advances to Cloud Services through Bentley's AssetWise Managed Services*. Atlanta

Abbas, S. S., Jayaprakash, P. O., Anitha, M., and Jaini, X. V. (2017) *Crowd Detection and Management using Cascade Classifier on ARMv8 and OpenCV-Python*.

Aitken, A. (2018) *Industry 4.0 Demystifying the Digital Twin* [online] available from <https://www.lanner.com/Assets/User/2511-Industry4.0_Demystifying_the_Digital_Twin.pdf> [February 2019]

Aitken, A. (2017) 'Getting Started with Industry 4.0: The Challenge for Many Businesses is Knowing Where to Begin.(SOFTWARE & ANALYSIS)'. *Quality* 56 (5), 30

Al-Bazi, A. (2017) *Introduction to Logistics Business Simulation*. Lecture module edn. UK: Coventry University

Allen, T. T. (2011a) *Introduction to Discrete Event Simulation and Agent-Based Modeling Voting Systems, Health Care, Military, and Manufacturing*. London: London : Springer

Allen, T. T. a. (2011b) *Introduction to Discrete Event Simulation and Agent-Based Modeling Voting Systems, Health Care, Military, and Manufacturing*.

Alnoman, A., Sharma, S., Ejaz, W., and Anpalagan, A. (2018) *Emerging Edge Computing Technologies for Distributed Internet of Things (IoT) Systems*. Ithaca

Austin, M. (2019) *Advantages of Plant Simulation and Siemens Digital Twin* [online] available from <https://www.attendmia.com/blog/2019/02/27/explore-the-latest-manufacturing-technology/> [04/10 2019]

B. P. Rimal, D. Pham Van, and M. Maier (2017) 'Mobile-Edge Computing Versus Centralized Cloud Computing Over a Converged FiWi Access Network'*. IEEE Transactions on Network and Service Management* 14 (3), 498-513

Babiceanu, R. F. and Seker, R. (2016) 'Big Data and Virtualization for Manufacturing Cyber-Physical Systems: A Survey of the Current Status and Future Outlook'. *Computers in Industry* 81, 128-137

Baran, P. (1964) *On Distributed Communications: I. Introduction to Distributed Communications Networks.* [online] available from <https://www.rand.org/pubs/research_memoranda/RM3420.html> [September 2018]

Beregi, R., Szaller, Á., and KáDáR, B. (2018) 'Synergy of Multi-Modelling for Process Control'. *IFAC PapersOnLine; IFAC PapersOnLine* 51 (11), 1023-1028

Bonomi, F., Milito, R., Zhu, J., and Addepalli, S. (eds.) (2012) *MCC '12 Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*. 'Fog Computing and its Role in the Internet of Things'. held 17/08/2012 at Helsinki, Finland

Brahmbhatt, S. a. (2013) *Practical OpenCV*.

Brailsford, S., editor, Churilov, L., editor, Dangerfield, B. T., editor, and Bayer, S., contributor (2014) *Discrete-Event Simulation and System Dynamics for Management Decision Making*.

Bruening, D. (2017) *Clustered/Stacked Filled Bar Graph Generator* [online] available from <http://www.burningcutlery.com/derek/bargraph/> [10/01 2018]

Byrne, J., Heavey, C., and Byrne, P. J. (2010) 'A Review of Web-Based Simulation and Supporting Tools'. *Simulation Modelling Practice and Theory* 18 (3), 253-276

Chen, W., author (2015) *Out-of-Order Parallel Discrete Event Simulation for Electronic System-Level Design*.

Chen, X., Henrickson, K., and Wang, Y. (2016) 'Kinect-Based Pedestrian Detection for Crowded Scenes.(Report)' 31 (3), 229

Chibuye, M. and Jackson, P. (2017) 'A Remote Sensor Network using Android Things and Cloud Computing for the Food Reserve Agency in Zambia'. *International Journal of Advanced Computer Science and Applications* 8 (11)

Ching-Tang Hsieh, Hui-Chun Wang, Yeh-Kuang Wu, Liung-Chun Chang, and Tai-Ku Kuo (2012) *A Kinect-Based People-Flow Counting System*.

Choi, B. K.,1949- author (2013) *Modeling and Simulation of Discrete Event Systems*.

Coskun, A., Kara, A., Parlaktuna, M., Ozkan, M., and Parlaktuna, O. (2015) *People Counting System by using Kinect Sensor*.

Croson, R. and Donohue, K. (2006) 'Behavioral Causes of the Bullwhip Effect and the Observed Value of Inventory Information'. *Management Science* 52 (3), 323-336

Csonto, J., Kadukova, J., and Polak, M. (2001) 'Artificial Life Simulation of Living Alga Cells and its Sorption Mechanisms'. *Journal of Medical Systems* 25 (3), 221-31

Eyre, J. and Freeman, C. (eds.) (2018) *EuroVR 2018*. 'Immersive Applications of Industrial Digital Twins'. held 2018/10/22 at London

Fabien, B. a. (2009) *Analytical System Dynamics Modeling and Simulation*.

Feng, Y. and Fan, W. (2013) 'A System Dynamics-Based Simulation Model of Production Line with Cross-Trained Workers'. *Journal of Statistical Computation and Simulation* 84 (6), 1-23

Frieß, U., Kolouch, M., Friedrich, A., and Zander, A. (2018) *Fuzzy-Clustering of Machine States for Condition Monitoring*.

Fujimoto, R. (1990) 'Parallel Discrete Event Simulation'. *Communications of the ACM* 33 (10), 30-53

Fu-Ren Lin, M. J., Gek, W. T., and Shaw, M. J. (1998) *Modeling Supply-Chain Networks by a Multi-Agent System*.

Georgiadis, P. and Michaloudis, C. (2011) 'Real-Time Production Planning and Control System for Job-Shop Manufacturing: A System Dynamics Analysis'. *European Journal of Operational Research* 216 (1)

Gilchrist, A., author (2016) *Industry 4.0 the Industrial Internet of Things*.

Gillham, B. (2000) *The Research Interview*. London: London : Continuum

Gómez, A., Cuiñas, D., Catalá, P., Xin, L., Li, W., Conway, S., and Lack, D. (2015) 'Use of Single Board Computers as Smart Sensors in the Manufacturing Industry'. *Procedia Engineering* 132, 153-159

Greaves, M. (2014) 'Digital Twin: Manufacturing Excellence through Virtual Factory Replication'

Guerreiro, B. V., Lins, R. G., Sun, J., and Schmitt, R. (2017) 'Definition of Smart Retrofitting: First Steps for a Company to Deploy Aspects of Industry 4.0. '. in *Advances in Manufacturing*. ed. by Hamrol A., Ciszak O., Legutko S., and Jurczyk M. : Springer, 161-170

Gupta, V., Narayanamurthy, G., and Acharya, P. (2018) *Can Lean Lead to Green? Assessment of Radial Tyre Manufacturing Processes using System Dynamics Modelling* [online] . available from <http://www.sciencedirect.com/science/article/pii/S0305054817300813>

H. Aydt, S. J. Turner, W. Cai, and M. Y. H. Low (eds.) (2009) *Proceedings of the 2009 Winter Simulation Conference (WSC)*. 'Research Issues in Symbiotic Simulation'

H. Derhamy, M. Andersson, J. Eliasson, and J. Delsing (eds.) (2018) *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*. 'Workflow Management for Edge Driven Manufacturing Systems'

Hibbert, L. (2014) 'Parallel Lines: A Virtual Version of Crossrail is Proving as Important to the London Rail Project's Smooth Running as its Physical Counterpart.(SOFTWARE)'. *Professional Engineering Magazine* 27 (1), 58

Hollocks, B. W. (2001) 'Discrete- Event Simulation: An Inquiry into User Practice'. *Simulation Practice and Theory* 8 (6), 451-471

Hou, B., Yao, Y., Wang, B., and Liao, D. (2013) 'Modeling and Simulation of Large-Scale Social Networks using Parallel Discrete Event Simulation'. *Simulation* 89 (10), 1173-1183

Hurwitz, J. (2013) *Big Data for Dummies*. 1st ed.. edn. Indianapolis, Ind.: Indianapolis, Ind. : John Wiley & Sons

Jianjun, N., Zubair, K., Shihao, W., Kang, W., and Syed, K. H. (eds.) (2016) *12th World Congress on Intelligent Control and Automation (WCICA)*. '**Automatic Detection and Counting of Circular Shaped Overlapped Objects using Circular Hough Transform and Contour Detection**'. held 12 June 2016 at Guilin, China: IEEE

Kahiomba, S. and Wang, Z. (2018) *Initiating Predictive Maintenance for a Conveyor Motor in a Bottling Plant using Industry 4.0 Concepts*. Heidelberg

Kehl, T. ***Agent-Based Modeling and Simulation with Simio*** [online] available from <https://www.simio.com/resources/papers/Agent-based-modeling-and-simulation-with-Simio.php> [September 2018]

Killingsworth, W. R. (2011) *Design, Analysis, and Optimization of Supply Chains a System Dynamics Approach*. 1st ed.. edn. New York, N.Y.] (222 East 46th Street, New York, NY 10017): New York, N.Y. 222 East 46th Street, New York, NY 10017 : Business Expert Press

Korth, B., Schwede, C., and Zajac, M. (2018) *Simulation-Ready Digital Twin for Realtime Management of Logistics Systems*.

Kritzinger, W., Karner, M., Traar, G., Henjes, J., and Sihn, W. (2018) *Digital Twin in Manufacturing: A Categorical Literature Review and Classification* [online] . available from <http://www.sciencedirect.com/science/article/pii/S2405896318316021>

Kun, A. J. and Vamossy, Z. (2009) *Traffic Monitoring with Computer Vision*.

Lapira, E. R. (2012) *Fault Detection in a Network of Similar Machines using Clustering Approach*. [online] thesis or dissertation: University of Cincinnati

Law, A. M. a. (2015) *Simulation Modeling and Analysis*. Fifth edition. International student edition.. edn

Lee, J., Lapira, E., Bagheri, B., and Kao, H. -. (2013) 'Recent Advances and Trends in Predictive Manufacturing Systems in Big Data Environment'. *Manufacturing Letters* 1 (1), 38-41

Lee, J., Bagheri, B., and Kao, H. (2015) 'A Cyber-Physical Systems Architecture for Industry 4.0-Based Manufacturing Systems'. *Manufacturing Letters* 3, 18-23

Leeming, M. (ed.) *Applied Visualisation Forum 2019*. 'THe Rise and Rise of the Digital Twin'. held 30 Jan 2019 at Gaydon, UK. https://tv.theiet.org/?eventvideoid=12739: The Institute of Engineering and Technology

Li, G., Ren, P., Lyu, X., and Zhang, H. (2016) *Real-Time Top-View People Counting Based on a Kinect and NVIDIA Jetson TK1 Integrated Platform*.

Liu, C. and Jiang, P. (2016) 'A Cyber-Physical System Architecture in Shop Floor for Intelligent Manufacturing'. *Procedia CIRP* 56, 372-377

McDonnel, R. (2013) *How to Manage Big Data from an Analog World* [online] available from <https://www.electronicdesign.com/analog/how-manage-big-data-analog-world> [10/01 2018]

Medhi, J. (. (2003) *Stochastic Models in Queueing Theory*. 2nd ed.. edn. Amsterdam; Boston: Amsterdam; Boston : Academic Press

Mhetre, A. (2018) 'Key Drivers and Benefits of Edge Computing for Smart Manufacturing: Edge Computing Means Faster Response Times, Increased Reliability, and Security. Five Edge Computing Advantages are Highlighted.(ANSWERS EDGE COMPUTING)'. *Control Engineering* 65 (3), 36

Murray, C. (1997) *Dawn of the Smart Conveyor*. Manhassett

Nightingale, R. (2017) *How does Google Maps Work?* [online] available from <https://www.makeuseof.com/tag/technology-explained-google-maps-work/> [10,1 2018]

Nikolakis, N., Alexopoulos, K., Xanthakis, E., and Chryssolouris, G. (2018) 'The Digital Twin Implementation for Linking the Virtual Representation of Human-Based Production Tasks to their Physical Counterpart in the Factory-Floor'. *International Journal of Computer Integrated Manufacturing* 32 (1), <xocs:first xmlns:xocs=""/>

Nilsson, R. (2013) 'Industrial Wireless: Bluetooth can be Robust, Easy to use'. *Computers--Automation, Engineering*

Nilsson, N. J., 1933- (2010) *The Quest for Artificial Intelligence : A History of Ideas and Achievements*.

Omar, M., Hussain, K., and Wright, S. (1999) 'Simulation Study of an Agile High-Speed Machining System for Automotive Cylinder Heads'. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 213 (5), 491-499

Peplow, M. (2016) *London's Crossrail is a $21 Billion Test of Virtual Modeling* [online] available from <https://spectrum.ieee.org/transportation/mass-transit/londons-crossrail-is-a-21-billion-test-of-virtual-modeling> [10/01 2018]

Pidd, M. (2004) *Computer Simulation in Management Science*. 5th ed.. edn. Chichester: Chichester : John Wiley & Sons

Qi, Q. and Tao, F. (2018) 'Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison'. *IEEE Access* 6, 3585-3593

Raggett, D. (2015) 'The Web of Things: Challenges and Opportunities'. *Computer* 48 (5), 26-32

Reinders, J. (2018) *Digital Twins: A Compelling use for Simulations on IoT Data* [online] available from <https://www.infoworld.com/article/3209761/digital-twins-a-compelling-use-for-simulations-on-iot-data.html> [10/01 2018]

Rivera-Gómez, H., Gharbi, A., and Kenné, J. P. (2013) *Joint Production and Major Maintenance Planning Policy of a Manufacturing System with Deteriorating Quality* [online] . available from <http://www.sciencedirect.com/science/article/pii/S0925527313003447>

Robinson, S.,1964- author (2014) *Simulation : The Practice of Model Development and use*. 2nd edition.. edn

Robinson, S. (2011) *Conceptual Modeling for Discrete-Event Simulation*. Boca Raton, FL: Boca Raton, FL : Taylor & Francis

Rosen, R., von Wichert, G., Lo, G., and Bettenhausen, K. D. (2015) *About the Importance of Autonomy and Digital Twins for the Future of Manufacturing* [online] . available from <http://www.sciencedirect.com/science/article/pii/S2405896315003808>

Rs Chen (2003) *Intelligent Warehousing Management Systems using Multi-Agent*.

Ruiz, N., Giret, A., Botti, V., and Feria, V. (2011) 'Agent-Supported Simulation Environment for Intelligent Manufacturing and Warehouse Management Systems'*. International Journal of Production Research* 49 (5), 1469-1482

S. Bergmann, S. Stelzer, and S. Straßburger (eds.) (2011) *Proceedings of the 2011 Winter Simulation Conference (WSC)*. 'Initialization of Simulation Models using CMSD'

Sajadi, S. M., Esfahani, M. M. S., and Sörensen, K. (2011) 'Production Control in a Failure-Prone Manufacturing Network using Discrete Event Simulation and Automated Response Surface Methodology'*. International Journal of Advanced Manufacturing Technology* 53 (1-4), 35-46

Sassi Hidri, M., Zoghlami, M. A., and Ben Ayed, R. (2018) *Speeding Up the Large-Scale Consensus Fuzzy Clustering for Handling Big Data* [online] . available from <http://www.sciencedirect.com/science/article/pii/S0165011417303901>

Saunders, M. N. K. (2016) *Research Methods for Business Students*. Seventh edition.. edn

Schaefer, D. and Cheung, W. M. (2018) 'Smart Packaging: Opportunities and Challenges'*. Procedia CIRP; Procedia CIRP* 72, 1022-1027

Schroeder, G. N., Steinmetz, C., Pereira, C. E., and Espindola, D. B. (2016) *Digital Twin Data Modeling with AutomationML and a Communication Methodology for Data Exchange* [online] . available from <http://www.sciencedirect.com/science/article/pii/S2405896316325538>

Seenouvong, N., Watchareeruetai, U., Nuthong, C., Khongsomboon, K., and Ohnishi, N. (2016) *A Computer Vision Based Vehicle Detection and Counting System*.

Sharda, B. and Bury, S. J. (2011) *Best Practices for Effective Application of Discrete Event Simulation in the Process Industries*.

Simio LLC (2017a) *Simulation Based Scheduling Software in Industry 4.0* [online] available from <https://www.simio.com/applications/industry-40/Simulation-Based-Scheduling-software-in-Industry-40.php> [10,01 2018]

Simio LLC (2017b) *Simio Simulation and Scheduling Software API Overview*. USA: Simio LLC

Simul8 (2018) *Planning for Industry 4.0 with Simulation* [online] available from <https://www.simul8.com/manufacturing/implementing-industry-4-0-with-simulation> [10/01 2018]

Suarez, O. D. (2014) *OpenCV Essentials : Acquire, Process, and Analyze Visual Content to Build Full-Fledged Imaging Applications using OpenCV*.

Suryatali, A. and Dharmadhikari, V. B. (2015) *Computer Vision Based Vehicle Detection for Toll Collection System using Embedded Linux*.

Taylor, M. (2018) *Crossrail Project: Building a Virtual Version of London's Elizabeth Line*. https://learninglegacy.crossrail.co.uk/wp-content/uploads/2017/09/12F-001-Building-a-virtual-version-of-Londons-Elizabeth-line.pdf edn. UK: Institute of Civil Engeers

Thiede, S., Juraschek, M., and Herrmann, C. (2016) 'Implementing Cyber-Physical Production Systems in Learning Factories'. *Procedia CIRP* 54, 7-12

Tocher, K. D. (1967) 'The Art of Simulation'*. English Universities Press*

Trinks, S. and Felden, C. (2018) *Edge Computing Architecture to Support Real Time Analytic Applications : A State-of-the-Art within the Application Area of Smart Factory and Industry 4.0*.

Vachalek, J., Bartalsky, L., Rovny, O., Sismisova, D., Morhac, M., and Loksik, M. (2017) *The Digital Twin of an Industrial Production Line within the Industry 4.0 Concept*.

W. Shi and S. Dustdar (2016) 'The Promise of Edge Computing'*. Computer* 49 (5), 78-81

Weber, A. (2018) *Industry 4.0 Drives Conveyor Technology*. Troy

Weber, A. (2016) 'Industry 4.0: Myths Vs. Reality'*. Assembly* 59 (12), 28-37

Whitt, W. (1991) 'The Efficiency of One Long Run Versus Independent Replications in Steady-State Simulation'*. Management Science* 37 (6), 645

Whyte, J. (2019) 'How Digital Information Transforms Project Delivery Models'*. Project Management Journal*, 8756972818823304

YANG, W., TAN, Y., and YOSHIDA, K. & TAKAKUWA, S. (2017) 'DIGITAL TWIN-DRIVEN SIMULATION FOR A CYBER-PHYSICAL SYSTEM IN INDUSTRY 4.0'*. DAAAM INTERNATIONAL SCIENTIFIC BOOK 2017*, 227-234

Yin, Y., Stecke, K. E., and Li, D. (2018) 'The Evolution of Production Systems from Industry 2.0 through Industry 4.0'*. International Journal of Production Research* 56 (1-2), 848-861

Zankoul, E., Khoury, H., and Awwad, R. (2015) *Evaluation of Agent-Based and Discrete-Event Simulation for Modeling Construction Earthmoving Operations* [online] available from <https://www.anylogic.com/upload/iblock/521/521fbe70be77461787e74b5ea216e468.pdf> [September 2018]

Zeigler, B. P. and Oren, T. I. (1979) 'Theory of Modelling and Simulation'*. Systems, Man and Cybernetics, IEEE Transactions on* 9 (1), 69-69

Zhuang, C., Liu, J., and Xiong, H. (2018) *Digital Twin-Based Smart Production Management and Control Framework for the Complex Product Assembly Shop-Floor*. s00170-018-1617-6 edn. US: The International Journal of Advanced Manufacturing Technology

Zio, E. a. (2013) *The Monte Carlo Simulation Method for System Reliability and Risk Analysis*.

Zupick, N. (2016) *Solving Agent Based Problems with Arena* [online] available from <https://www.arenasimulation.com/blog/post/solving-agent-based-problems-with-arena> [September 2018]

Smart Factory: Developing a Digital Twin and Intelligent Network for Self-Adaptive, Flexible Production Environments

Book of Appendices

Coventry University

Author: Simon Rollinson

Appendix 1

Ethical Application Document

Certificate of Ethical Approval

Applicant:

Simon Rollinson

Project Title:

Smart Factory: developing cyber physical systems and intelligent networks for self-adaptive, flexible production environments

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Medium Risk

Date of approval:

13 April 2018

Project Reference Number:

P61796

Appendix 2
Braunstone Specification document

Author: Simon Rollinson

Date of release: 18/02/2018

Release no: 1

Authority:

Introduction to the problem

The food can production line at Braunstone manufactures 2-piece food cans by drawing a cup and then re-drawing and wall ironing the cup in the Bodymaker to the size and specification of the can. The Jefferson process sits between the Cupper and Bodymaker to redraw the cup in order to thin the metal at the base of a 2 piece food can, thus saving a small amount of metal. The machine is expected to increase the yield from a coil of steel from 600,000 cans to 636,000 cans.

The objective of the simulation is to determine how many Jefferson machines are required so as to maintain current production level.

Project objectives

The following objectives have been agreed:

1. Baseline model and validate against appropriate production data.
2. Jefferson Capacity from 5 to 10 machines

Expected Benefits

- A better understanding of the *Production Process* by *Crown Technology*

- A better understanding of DES projects by Crown Food Braunstone

- Quantification of the results of various changes that can be made to the Production Process, in terms of altering the values of input factors (in isolation and in combination) and the consequent effects on model outputs.

Scope

This sub section outlines the boundaries of the modelling process.

- In order to avoid an overly complex simulation model (and therefore waste time on modelling experimental factors that give irrelevant or too limited information), the model only needs to include machine line controls and neglect conveyor line controls

- To avoid long simulation time, the model will be run for a
  - Maximum of 45 days of production process time and results will be extrapolated from this.
  - Each model entity will represent between 2 and 50 cans, with exact scale factor determined experimentally.

Assumptions

The following details are assumed within the modelling process:

1. The de-reeler never runs out of stock.
2. The conveyors do not create hold ups.
3. Jefferson will operate at 200 cans per minute
4. Failure data for the Jefferson machine is the same as the Cupper
5. Scrap rates are not significant

General details

- Machine Operation: Cycle times for each machine have been modelled using the data provided see appendix 2.

- Machine breakdown/stoppage durations: (MTTR – Mean Time To Repair) and the frequency (MTBF – Mean Time Between Failure) have been modelled using the data from SMARTLINE *Failure Data*. Each machine dataset has been treated as an *exponential* distribution. Outlier data points within the data for each machine cycle time are included/excluded within the distributions, as these are/not considered significant events.

- Jefferson will operate at a rate of 200 cans per minute with failure data for the Jefferson machines equivalent to Cupper machine.

- No financial justifications will be included in the report.

Experimental factors

- Jefferson machine capacity from 5 to 10 will be modelled.

Model requirements

- The software to be used is Simio.

- Functionally, the software only needs to provide a relatively minimal level of graphical representation.

- Outputs from Simio will be assimilated within Microsoft Excel in order to convey results.

Description of Operation

The process flow/Activity Cycle Diagram of the process that is to be modelled is shown below:

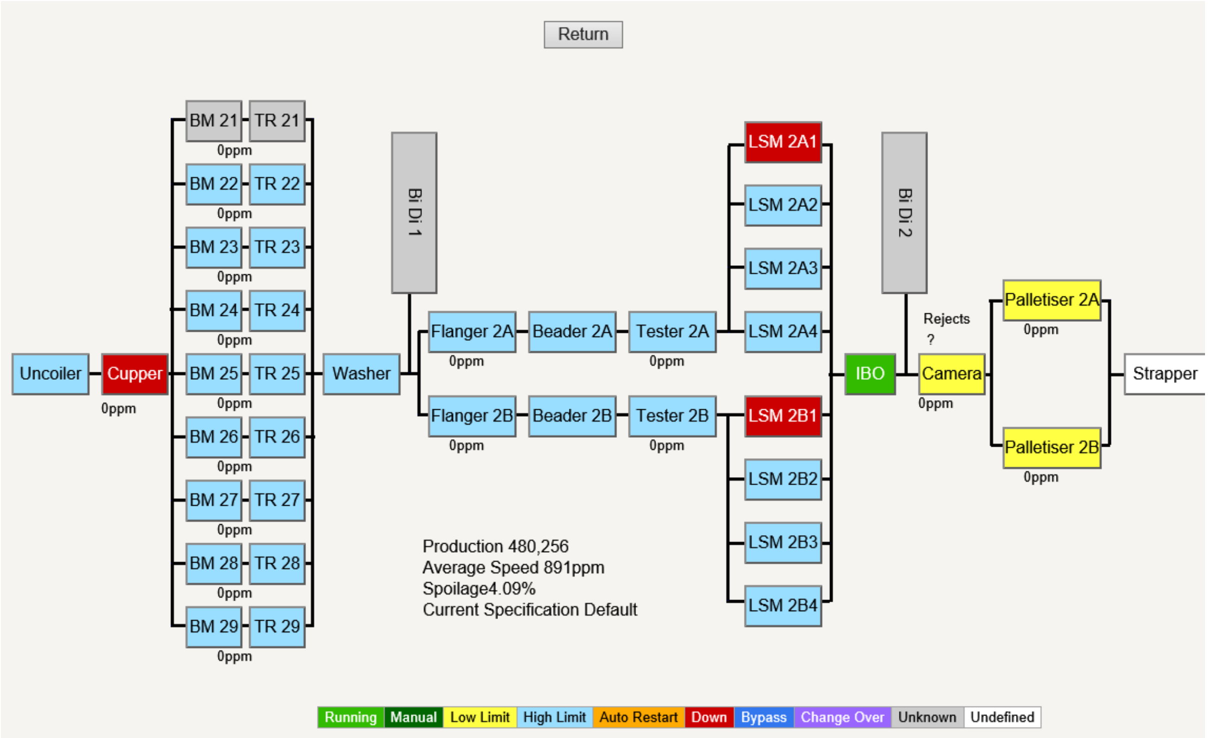


Figure 1 Smartline Mimic

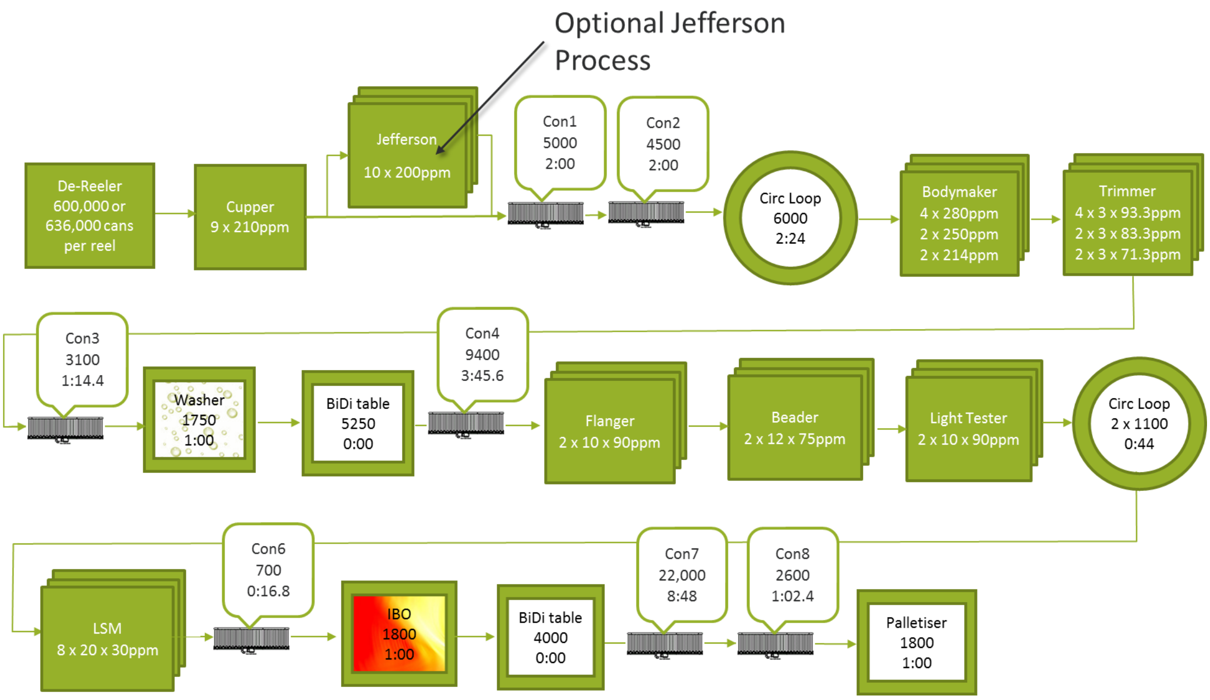


Figure 2 Outline Conceptual Model

Reports

- Throughput of Process, under each scenario.
- Recommendations based on a comparison of the original performance of Process with the experimental cases.

Project Management

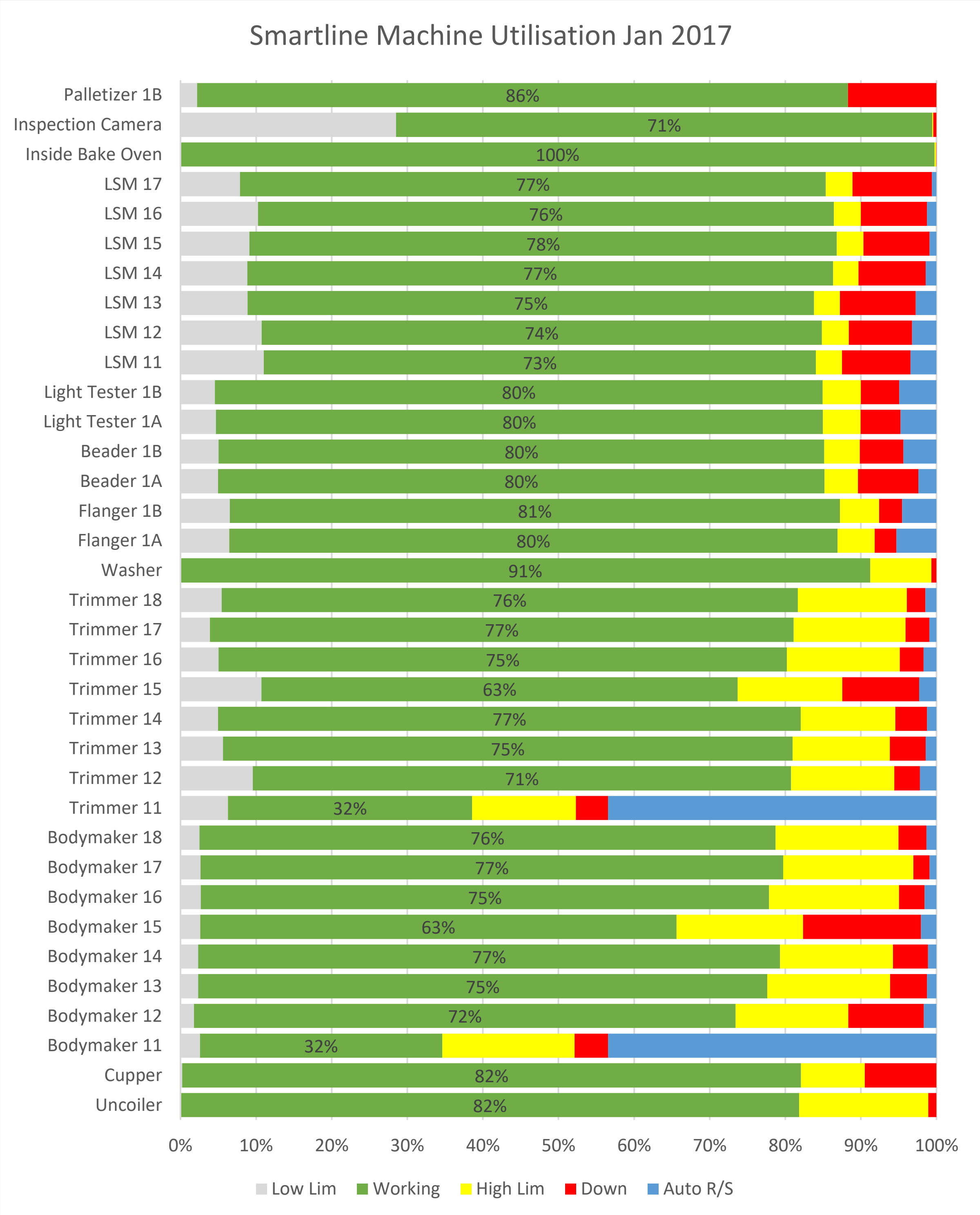
Simon Rollinson is the Project Manager. He will communicate directly with Crown Braunstone as required. The formal project plan is shown below, without dates:

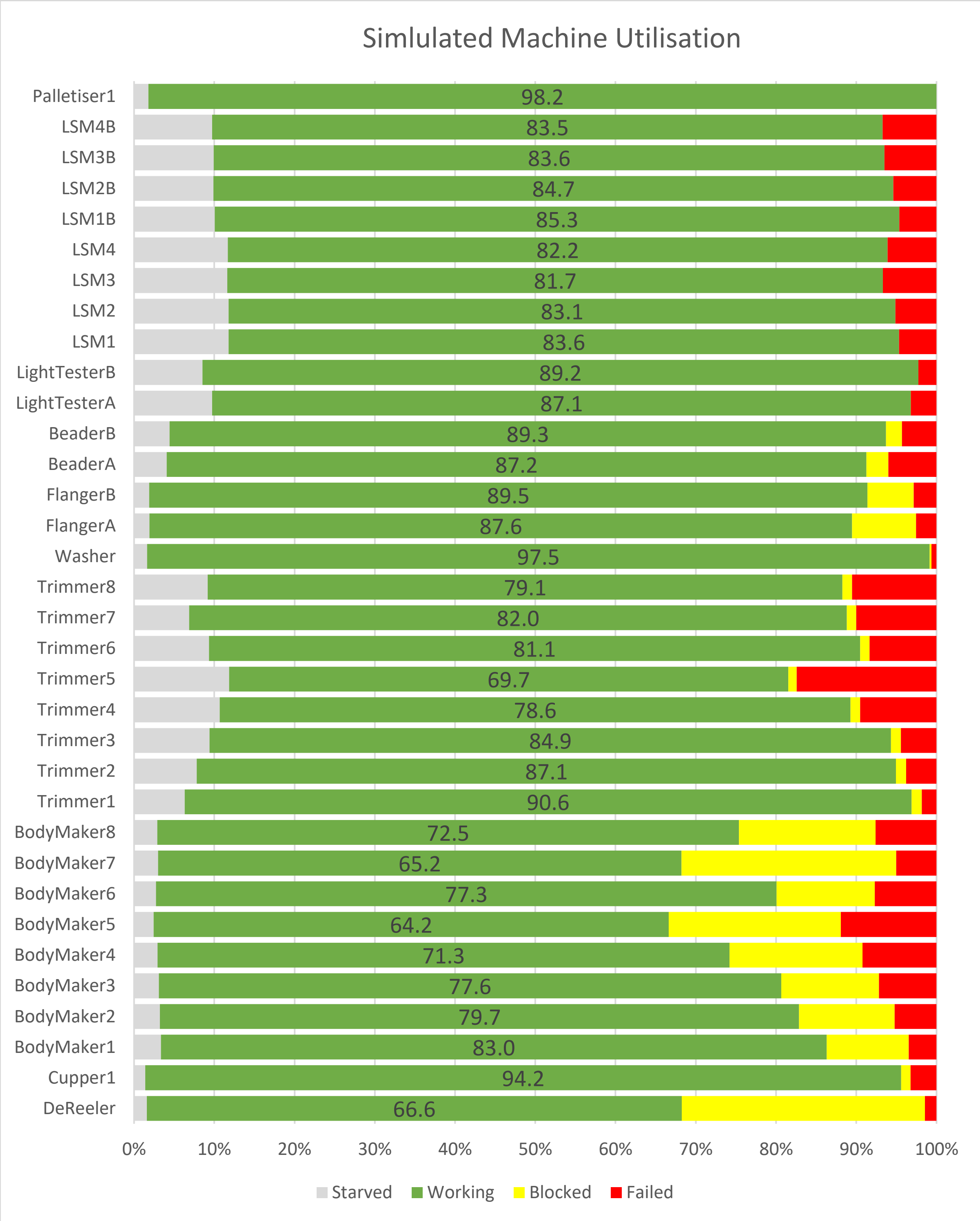
Task Number	Task Name	Duration	Predecessors	Resource Names
1	Visit 1 - Familiarisation	1 day		
2	Project Management	1 day		
3	Visit 2 - Data Collection	2 day	5	
4	Conceptualisation - ACD and Process Flow Diagram	5 day	1	
5	Specification Document	3 day	3	
6	Validation 1	1 day	3	
7	DES Modelling	5 days	6	
8	Model Validation	1 day	7	
9	Experimentation	4 days	8	
10	Results	1 day	9	
11	Documentation	2 days	10	
12	Refinement/Contingency	3 days		

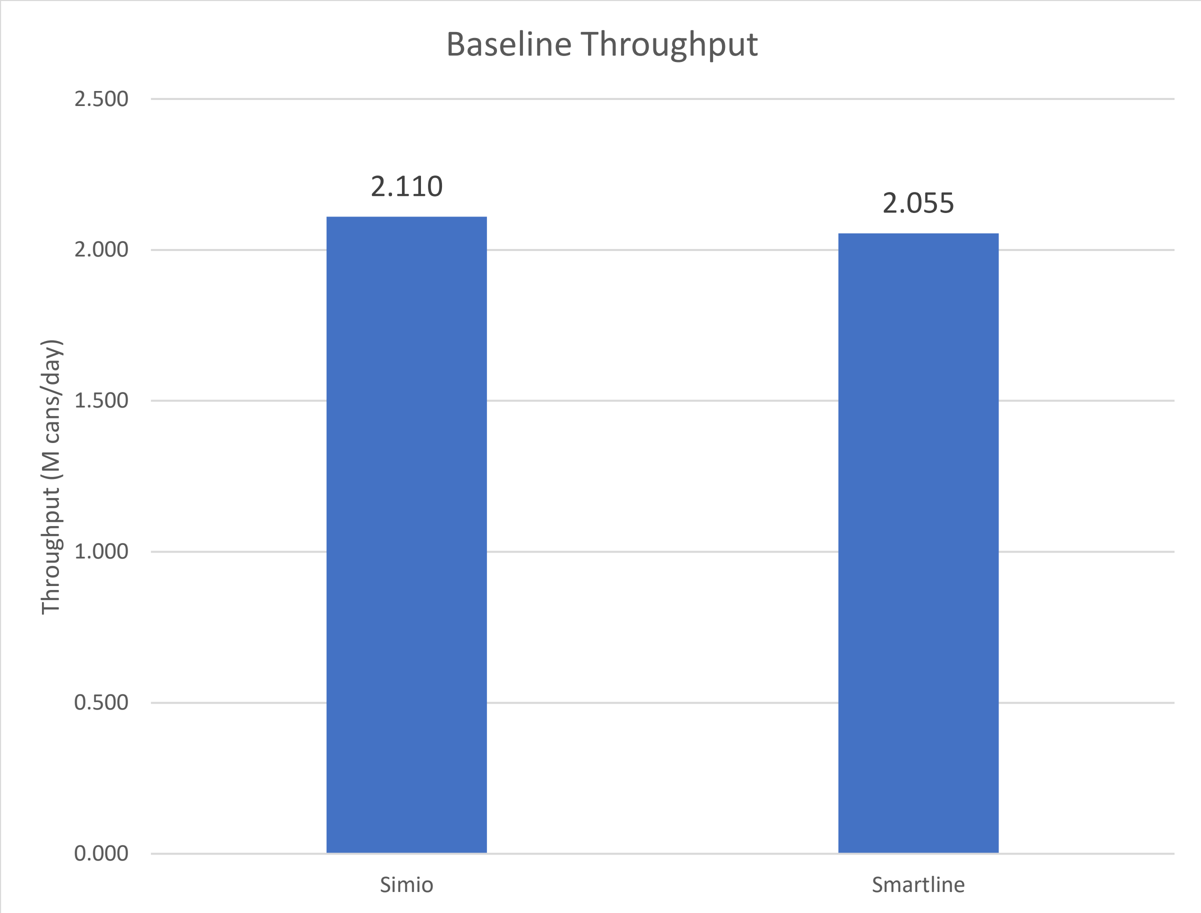
Results

Validation

The Smartline system shows average daily production at 2.05M cans per day (see appendix 1). The simulated output is about 2% higher.



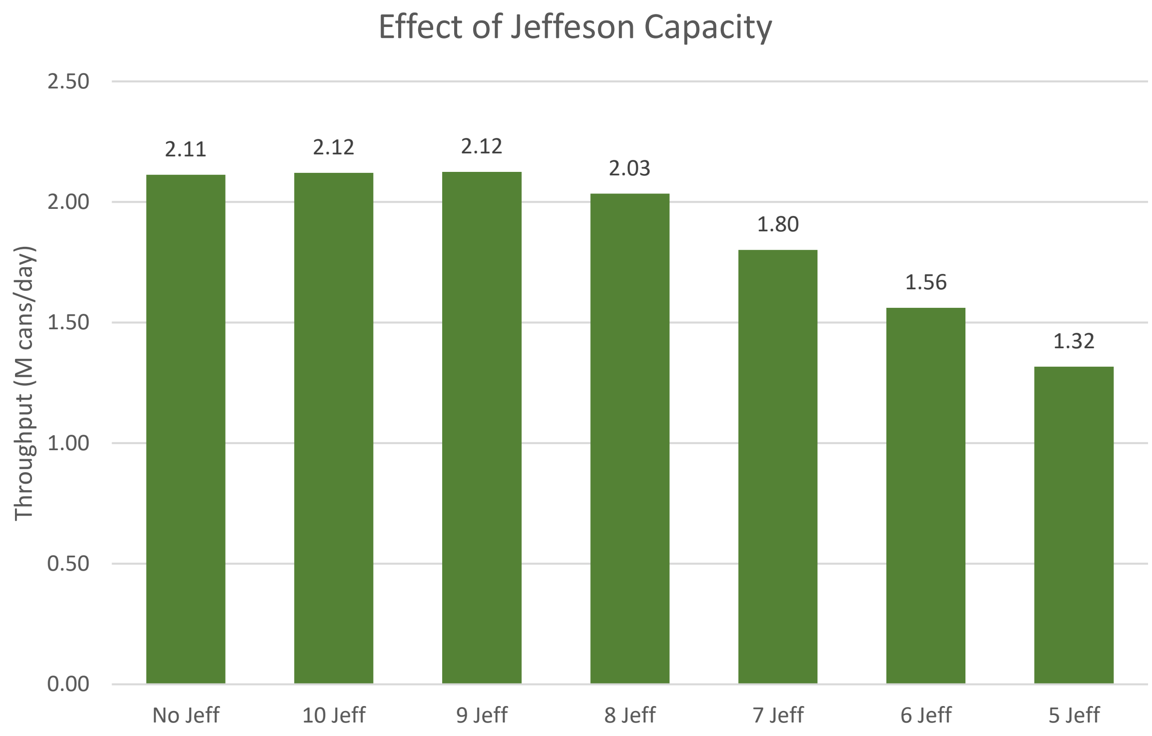

Simlulated Machine Utilisation
Palletiser1
LSM4B
LSM3B
LSM2B
LSM1B
LSM4
LSM3
LSM2
LSM1
LightTesterB
LightTesterA
BeaderB
BeaderA
FlangerB
FlangerA
Washer
Trimmer8
Trimmer7
Trimmer6
Trimmer5
Trimmer4
Trimmer3
Trimmer2
Trimmer1
BodyMaker8
BodyMaker7
BodyMaker6
BodyMaker5
BodyMaker4
BodyMaker3
BodyMaker2
BodyMaker1
Cupper1
DeReeler
98.2
83.5
83.6
84.7
85.3
82.2
81.7
83.1
83.6
89.2
87.1
89.3
87.2
89.5
87.6
97.5
79.1
82.0
81.1
69.7
78.6
84.9
87.1
90.6
72.5
65.2
77.3
64.2
71.3
77.6
79.7
83.0
94.2
66.6
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Starved
Working
Blocked
Failed



Jefferson Experiment

The introduction of the Jefferson machines, between the Cupper and Bodymakers.

The data shows that, if the assumptions about throughput and breakdowns are correct, that 9 Jefferson machines would be required to maintain production at current levels.



Conclusions

- The baseline model is a good estimation of the physical production line.
- 9 Jefferson machines would be required to maintain throughput at current levels.

Coventry University

CROWN Brand-Building Packaging™

Appendix 1 – Smartline Data

Department: D&I **Line:** 1

Start Date: 01/11/2017 **Shift:** Day (Standard)

End Date: 10/11/2017 **Shift:** Night (Standard)

				Down		Auto Restart		Low Limit		High Limit									
Charts	Machine	Production	Running Time	Count	Time	Count	Time	Count	Time	Count	Time	MTBF	MTTR	Efficiency (%)	Availability(%)	Spoilage (%)	Average Speed (ppm)	Total Time	Alarms
D&I Line 1																			
Total Production: 20,554,240																			
Total Spoilage: 0.27 (%)																			
Average Line Speed: 1,427 (ppm)																			
	Uncoiler		196:22:34	110	2:36:57					660	40:55:46	1:47:07	0:01:26		98.91			239:56:18	0
	Cupper	20610000	196:22:48	421	**22:43:25**			15	0:33:00	259	20:16:08	0:27:59	0:03:14	75.73	90.53		1749	239:56:22	0
	Bodymaker 11	937240	76:56:11	84	10:37:04	966	104:14:07	417	6:10:01	308	41:57:33	0:54:57	**0:07:35**	**23.25**	95.57	0.06	203	239:55:57	0
	Bodymaker 12	2872130	171:54:01	135	**24:00:23**	681	3:59:51	317	4:17:36	305	35:42:35	1:16:24	**0:10:40**	71.23	**89.99**	-0.05	278	239:55:27	0
	Bodymaker 13	3034300	180:36:44	109	11:38:19	620	3:03:01	270	5:36:55	289	38:59:38	1:39:25	**0:06:24**	75.26	95.15	-0.02	280	239:55:38	0
	Bodymaker 14	3067530	184:42:15	143	11:08:07	686	2:40:49	266	5:36:24	328	35:47:03	1:17:30	0:04:40	76.08	95.36	0.05	277	239:55:39	0
	Bodymaker 15	2272330	151:04:49	155	**37:26:20**	588	4:54:01	466	6:17:50	472	40:11:08	0:58:29	**0:14:30**	73.74	**84.39**	0.75	253	239:55:09	0
	Bodymaker 16	2733350	180:17:14	76	8:11:02	493	3:43:47	228	6:28:15	443	41:14:14	2:22:20	**0:06:28**	75.93	96.59	0.4	254	239:55:33	0
	Bodymaker 17	2786960	184:48:33	58	5:07:34	383	2:09:47	230	6:20:01	334	41:16:54	3:11:11	0:05:18	77.42	97.86	0.5	253	239:55:51	0
	Bodymaker 18	2848610	182:51:12	56	8:55:12	316	3:09:43	215	6:01:29	265	38:57:20	3:15:55	**0:09:33**	79.13	96.28	0.16	260	239:55:57	0
	Trimmer 11		77:30:58	63	10:09:57	1329	104:14:22	428	15:03:21	257	32:56:16	1:13:49	**0:09:41**		95.76			239:55:55	0
	Trimmer 12		170:53:12	98	8:09:12	1032	5:15:30	500	22:54:24	279	32:41:59	1:44:37	0:05:00		96.6			239:55:18	0
	Trimmer 13		180:46:20	85	11:18:50	696	3:26:48	364	13:28:35	237	30:53:52	2:07:36	**0:07:59**		95.28			239:55:26	0
	Trimmer 14		184:54:54	118	10:03:50	746	3:01:19	423	11:55:21	250	29:59:10	1:34:01	0:05:07		95.81			239:55:35	0
	Trimmer 15		151:10:10	279	**24:22:55**	1337	5:29:45	528	25:40:51	270	33:10:26	0:32:31	0:05:15		**89.84**			239:55:08	0
	Trimmer 16		180:24:18	201	7:33:31	1010	4:06:04	315	12:03:36	284	35:46:51	0:53:51	0:02:15		96.85			239:55:21	0
	Trimmer 17		185:04:54	116	7:29:43	726	2:17:23	290	9:20:53	239	35:29:52	1:35:44	0:03:53		96.87			239:55:47	0
	Trimmer 18		182:50:16	53	5:52:45	757	3:32:40	290	13:02:35	234	34:36:23	3:26:59	**0:06:39**		97.55			239:55:40	0
	Washer		218:51:53	18	1:35:08					215	19:28:19	12:09:33	0:05:17		99.34			239:56:21	0
	Flanger 1A	7182340	193:00:31	236	6:52:44	956	12:45:21	199	15:28:43	540	11:46:51	0:49:04	0:01:45	**55.42**	97.13	**29.32**	878	239:55:11	0
	Flanger 1B	10815640	193:38:45	457	7:19:43	1213	10:53:15	209	15:38:41	596	12:24:02	**0:25:25**	0:00:58	83.45	96.95	-3.08	903	239:55:27	0
	Beader 1A		192:28:38	330	19:11:37	1326	5:43:19	202	11:55:51	559	10:34:37	0:35:00	0:03:29		92			239:55:03	0
	Beader 1B		192:14:39	389	13:51:15	1644	10:28:15	215	12:04:35	511	11:15:27	0:29:39	0:02:08		94.23			239:55:12	0
	Light Tester 1A	0	192:36:47	154	12:42:12	1337	11:25:20	210	11:13:03	609	11:56:49	1:15:03	0:04:57	**0**	94.7		0	239:55:12	0
	Light Tester 1B	0	192:54:02	206	12:14:51	1718	11:46:46	199	10:53:55	614	12:04:31	0:56:11	0:03:34	**0**	94.89		0	239:55:06	0
	Lacquer Spray Machine 11	2815346	175:09:51	246	**21:48:15**	1728	8:12:25	2681	26:27:57	163	8:16:04	0:42:43	0:05:19	61.1	90.91		268	239:54:32	0
	Lacquer Spray Machine 12	2856950	177:46:51	232	**20:05:14**	2114	7:44:25	3335	25:44:19	168	8:33:34	0:45:59	0:05:12	62	91.63		268	239:54:23	0
	Lacquer Spray Machine 13	2891797	179:45:00	283	**24:06:11**	1744	6:34:38	3007	21:19:35	161	8:08:26	0:38:07	0:05:07	62.76	89.95		268	239:53:50	0
	Lacquer Spray Machine 14	3007112	185:52:18	309	**21:21:12**	1277	3:23:22	2821	21:12:47	158	8:04:13	0:36:05	0:04:09	65.26	91.1		270	239:53:52	0
	Lacquer Spray Machine 15	3018874	186:19:38	218	**20:56:35**	627	2:12:51	3832	21:54:32	161	8:30:03	0:51:17	0:05:46	65.51	91.27		270	239:53:39	0
	Lacquer Spray Machine 16	2953370	182:45:59	221	**21:01:25**	1298	3:00:41	3215	24:34:07	164	8:31:54	0:49:37	0:05:42	64.09	91.24		269	239:54:06	0
	Lacquer Spray Machine 17	3017593	185:52:24	307	**25:16:26**	974	1:25:02	2318	18:54:40	158	8:25:32	0:36:20	0:04:56	65.49	89.46		271	239:54:04	0
	Inside Bake Oven		239:19:10	5	0:06:49					43	0:29:11	47:51:50	0:01:22		99.95			239:56:11	0
	Inspection Camera	20389104	170:06:35	44	1:03:11			14772	68:26:27	9	0:18:46	3:51:58	0:01:26	**78.66**	99.56		1999	239:56:00	17
	Palletizer 1A	0		2	**239:55:18**								**119:57:39**	**0**	**0**			239:56:19	0
	Palletizer 1B	20554240	206:32:13	940	**28:05:24**			65	5:17:14			**0:13:11**	0:01:48	**62.06**	**88.29**		1659	239:55:52	0

Coventry University

CROWN
Brand-Building Packaging™

Department:	D&I	**Line:**	1
Start Date:	01/10/2017	**Shift:**	Day (Standard)
End Date:	10/10/2017	**Shift:**	Night (Standard)

				Down		Auto Restart		Low Limit		High Limit									
Charts	Machine	Production	Running Time	Count	Time	Count	Time	Count	Time	Count	Time	MTBF	MTTR	Efficiency (%)	Availability(%)	Spoilage (%)	Average Speed (ppm)	Total Time	Alarms
D&I Line 1																			
Total Production: 20,130,624																			
Total Spoilage: 0.76 (%)																			
Average Line Speed: 1,398 (ppm)																			
	Uncoiler		194:15:41	97	2:10:50					650	43:00:30	2:00:10	0:01:21		99.09			239:56:23	0
	Cupper	20285190	194:15:47	402	28:55:38			6	0:04:10	264	16:11:31	0:29:00	0:04:19	74.61	87.92		1740	239:56:28	0
	Bodymaker 11	1108980	91:05:38	43	28:25:56	701	86:09:16	429	7:09:46	195	26:25:19	2:07:06	0:39:40	27.53	88.12	0.26	203	239:55:43	0
	Bodymaker 12	2959000	185:21:42	113	14:19:24	685	3:17:02	342	4:02:47	264	32:25:20	1:38:25	0:07:36	73.46	94.02	-0.01	266	239:55:37	0
	Bodymaker 13	2792330	170:37:24	303	35:28:30	611	2:48:03	245	5:03:38	201	24:31:12	0:33:47	0:07:01	69.33	85.12	-0.01	273	239:55:40	0
	Bodymaker 14	2859320	171:50:57	240	32:06:16	571	6:04:10	181	3:55:40	221	25:29:37	0:42:58	0:08:02	70.99	86.59	0.01	277	239:56:02	0
	Bodymaker 15	2473560	165:21:25	197	29:54:02	543	3:45:23	267	6:03:02	382	34:22:42	0:50:22	0:09:06	80.35	87.51	0.43	250	239:55:56	0
	Bodymaker 16	2632730	173:43:29	96	15:58:38	514	3:02:44	195	5:52:00	450	40:49:37	1:48:35	0:09:59	73.21	93.33	0.57	254	239:55:50	0
	Bodymaker 17	2708770	179:48:14	82	16:42:33	370	3:37:38	226	5:30:59	309	33:46:55	2:11:34	0:12:14	75.32	93.02	0.77	253	239:55:41	0
	Bodymaker 18	2687260	172:22:57	85	22:22:25	300	2:37:28	190	5:24:44	221	36:38:55	2:01:41	0:15:48	74.72	90.66	0.2	260	239:55:51	0
	Trimmer 11		91:36:35	57	14:46:37	1099	86:12:06	380	21:58:57	178	24:41:52	1:36:26	0:15:33		93.82			239:55:55	0
	Trimmer 12		178:02:23	101	19:24:05	892	3:29:56	498	12:34:14	195	25:55:40	1:45:46	0:11:32		91.9			239:55:40	0
	Trimmer 13		170:28:37	65	10:30:55	800	3:30:19	455	28:51:01	179	25:07:50	2:37:22	0:09:42		95.59			239:55:35	0
	Trimmer 14		172:00:55	72	9:14:00	608	6:18:56	449	26:51:37	175	25:01:10	2:23:21	0:07:42		96.14			239:56:00	0
	Trimmer 15		165:26:38	242	24:55:45	1023	4:09:38	485	19:02:49	203	25:51:23	0:41:01	0:06:11		89.59			239:55:35	0
	Trimmer 16		173:48:58	291	22:50:28	994	3:22:35	311	13:12:22	213	26:11:51	0:35:50	0:04:43		90.46			239:55:36	0
	Trimmer 17		180:02:32	158	16:21:22	657	3:46:08	301	16:00:23	168	23:15:53	1:08:22	0:06:13		93.17			239:55:40	0
	Trimmer 18		172:22:08	76	19:58:53	695	2:58:26	294	19:12:10	168	24:54:49	2:16:05	0:15:46		91.65			239:55:48	0
	Washer		221:58:22	8	2:16:03					157	15:12:35	27:44:48	0:17:00		99.05			239:56:22	0
	Flanger 1A	6969140	192:56:23	404	12:20:56	991	10:50:53	217	12:46:20	828	10:30:54	0:28:39	0:01:50	53.83	94.84	31.2	875	239:54:48	0
	Flanger 1B	10037180	188:50:59	713	15:15:36	1536	12:27:14	211	11:42:57	860	11:08:10	0:15:54	0:01:17	77.53	93.63	1.81	902	239:54:18	0
	Beader 1A		192:16:06	312	15:55:26	1647	8:09:03	219	12:03:09	906	11:01:46	0:36:58	0:03:04		93.35			239:54:52	0
	Beader 1B		187:02:13	360	15:24:48	2031	14:06:26	230	12:38:10	762	10:13:29	0:31:10	0:02:34		93.56			239:54:28	0
	Light Tester 1A	0	192:11:37	122	18:38:06	1510	8:50:50	217	8:16:05	936	11:28:55	1:34:31	0:09:10	0	92.22		0	239:54:55	0
	Light Tester 1B	0	187:40:01	250	19:35:39	2088	12:27:08	227	8:21:20	883	11:21:10	0:45:02	0:04:42	0	91.82		0	239:54:40	0
	Lacquer Spray Machine 11	2802991	171:47:46	341	34:32:19	1330	3:58:59	2473	23:41:39	202	5:23:15	0:30:14	0:06:05	60.89	85.57		272	239:53:56	0
	Lacquer Spray Machine 12	2754042	169:27:34	296	35:27:31	2042	4:13:18	3674	24:53:41	211	5:22:07	0:34:21	0:07:11	59.83	85.19		271	239:54:09	0
	Lacquer Spray Machine 13	2946942	180:05:31	390	32:05:56	1543	3:35:55	3364	18:23:35	201	5:12:57	0:27:42	0:04:56	64.02	86.59		273	239:53:52	0
	Lacquer Spray Machine 14	2974885	181:47:52	477	35:13:36	1224	2:09:12	2893	15:21:20	204	4:51:53	0:22:52	0:04:26	64.63	85.29		273	239:53:51	0
	Lacquer Spray Machine 15	3035260	184:18:11	288	30:55:41	467	1:07:07	3017	17:15:28	217	5:47:22	0:38:24	0:06:27	65.94	87.08		274	239:53:47	0
	Lacquer Spray Machine 16	2818027	172:51:10	574	45:11:41	1178	1:31:27	2837	14:28:41	210	5:21:03	0:18:04	0:04:43	61.22	81.12		272	239:54:00	0
	Lacquer Spray Machine 17	2881988	175:44:33	355	44:13:27	882	1:01:14	2184	13:47:42	190	4:36:35	0:29:42	0:07:28	62.61	81.53		273	239:53:29	0
	Inside Bake Oven		237:30:32	7	0:44:23					95	1:11:48	33:55:47	0:06:20		99.69			239:56:05	0
	Inspection Camera	19878365	162:17:38	185	1:38:13			17823	75:04:06	10	0:26:49	0:52:38	0:00:32	76.77	99.32		2049	239:56:08	57
	Palletizer 1A	0		1	239:26:59								239:26:59	0	0			239:56:21	0
	Palletizer 1B	20130624	201:36:37	897	33:25:24			45	4:24:12			0:13:29	0:02:14	60.84	86.04		1664	239:55:35	0

Coventry University

CROWN
Brand-Building Packaging™

Department: D&I **Line:** 1
Start Date: 01/12/2017 **Shift:** Day (Standard)
End Date: 10/12/2017 **Shift:** Night (Standard)

				Down		Auto Restart		Low Limit		High Limit									
Charts	Machine	Production	Running Time	Count	Time	Count	Time	Count	Time	Count	Time	MTBF	MTTR	Efficiency (%)	Availability(%)	Spoilage (%)	Average Speed (ppm)	Total Time	Alarms
D&I Line 1																			
Total Production: 21,040,416																			
Total Spoilage: 0.56 (%)																			
Average Line Speed: 1,461 (ppm)																			
	Uncoiler		203:59:50	93	1:46:44					844	34:07:22	2:11:37	0:01:09		99.26			239:53:56	0
	Cupper	21159720	204:00:06	472	14:45:49			17	0:32:31	383	20:35:22	**0:25:56**	0:01:53	77.75	93.85		1729	239:53:48	0
	Bodymaker 11	1243830	101:58:48	70	5:59:24	1311	97:16:14	714	5:41:56	376	28:58:12	1:27:25	0:05:08	**30.85**	97.5	0.08	203	239:54:34	0
	Bodymaker 12	3104170	184:22:02	200	15:58:32	844	7:05:50	271	2:12:01	455	30:14:44	0:55:19	0:04:48	76.99	93.34	-0.06	280	239:53:09	0
	Bodymaker 13	3096660	184:40:31	215	15:48:41	672	7:41:00	236	3:14:29	363	28:28:28	0:51:32	0:04:25	76.8	93.41	-0.01	279	239:53:09	0
	Bodymaker 14	3026100	181:59:34	220	14:39:10	795	6:25:57	214	3:11:32	380	33:36:46	0:49:38	0:04:00	75.05	93.89	0.01	277	239:52:59	0
	Bodymaker 15	2550890	170:27:02	292	**24:31:38**	849	6:03:57	343	4:35:46	611	34:15:27	0:35:01	0:05:02	82.78	**89.78**	1.36	253	239:53:50	0
	Bodymaker 16	2813350	186:19:23	79	9:52:43	631	7:30:07	195	4:15:27	561	31:55:47	2:21:31	**0:07:30**	78.15	95.88	0.86	254	239:53:27	0
	Bodymaker 17	2241440	149:36:20	154	**48:07:52**	678	4:08:52	195	3:18:00	548	34:42:20	0:58:17	**0:18:45**	**62.26**	**79.94**	1.28	253	239:53:24	0
	Bodymaker 18	2974360	191:14:55	161	13:21:22	553	3:01:47	215	3:41:11	408	28:34:21	1:11:16	0:04:59	82.62	94.43	0.39	260	239:53:36	0
	Trimmer 11		102:52:26	23	2:34:30	1930	97:24:11	519	8:49:31	361	28:13:52	4:28:22	**0:06:43**		98.93			239:54:30	0
	Trimmer 12		184:31:06	147	9:10:26	1210	7:16:24	493	11:04:50	374	27:50:26	1:15:19	0:03:45		96.18			239:53:12	0
	Trimmer 13		184:24:47	71	14:48:08	903	8:34:11	339	7:37:20	325	24:28:41	2:35:51	**0:12:31**		93.83			239:53:07	0
	Trimmer 14		182:08:28	64	12:08:57	823	6:47:12	462	13:29:15	332	25:18:57	2:50:45	**0:11:23**		94.94			239:52:49	0
	Trimmer 15		170:34:12	374	18:33:18	1616	6:41:50	679	16:47:41	370	27:16:33	0:27:22	0:02:59		92.27			239:53:34	0
	Trimmer 16		186:28:58	242	8:10:24	1194	7:49:49	287	9:53:43	381	27:30:30	0:46:14	0:02:02		96.59			239:53:24	0
	Trimmer 17		149:57:00	325	**42:56:41**	1103	4:18:36	435	17:27:32	321	25:13:37	0:27:41	**0:07:56**		**82.1**			239:53:26	0
	Trimmer 18		191:12:34	110	6:15:55	1164	3:32:08	411	12:15:04	332	26:37:36	1:44:18	0:03:25		97.39			239:53:17	0
	Washer		219:44:27	6	0:11:30					309	19:57:24	36:37:25	0:01:55		99.92			239:53:21	0
	Flanger 1A	7164490	197:49:38	484	9:07:03	6201	18:53:05	157	6:36:48	334	7:26:21	**0:24:31**	0:01:08	**55.28**	96.2	**28.27**	841	239:53:07	2
	Flanger 1B	11248620	204:08:25	523	12:28:34	1101	8:35:34	158	6:37:05	407	8:03:44	**0:23:25**	0:01:26	86.79	94.8	-0.18	917	239:53:22	1
	Beader 1A		199:38:54	413	10:43:36	2349	16:21:48	164	6:49:22	356	6:19:08	0:29:00	0:01:34		95.53			239:53:00	0
	Beader 1B		203:02:13	366	11:56:24	1442	12:01:16	158	6:36:23	344	6:16:56	0:33:17	0:01:57		95.02			239:53:24	0
	Light Tester 1A	9931110	199:24:02	268	15:42:17	2348	12:09:47	159	5:56:08	377	6:40:46	0:44:39	0:03:31	76.63	93.45		834	239:53:12	0
	Light Tester 1B	0	203:49:01	157	7:28:09	1476	15:46:32	178	5:58:46	418	6:50:42	1:17:54	0:02:51	**0**	96.89		0	239:53:10	0
	Lacquer Spray Machine 11	2909145	185:34:26	266	16:09:30	1628	6:32:21	3065	27:23:12	139	4:14:46	0:41:52	0:03:39	63.13	93.26		261	239:54:15	0
	Lacquer Spray Machine 12	2901477	185:32:05	268	14:16:18	2198	6:28:36	3823	29:03:46	138	4:33:26	0:41:32	0:03:12	62.97	94.05		261	239:54:11	0
	Lacquer Spray Machine 13	3003266	190:59:32	283	15:23:22	1476	5:05:09	3071	24:20:26	128	4:05:31	0:40:30	0:03:16	65.18	93.59		262	239:54:00	0
	Lacquer Spray Machine 14	3062126	194:14:37	597	17:41:26	1144	2:49:46	2768	20:39:36	127	4:28:23	**0:19:31**	0:01:47	66.45	92.63		263	239:53:48	0
	Lacquer Spray Machine 15	3135138	198:00:55	231	12:41:08	491	0:56:38	3755	23:21:40	134	4:53:35	0:51:26	0:03:18	68.04	94.71		264	239:53:56	0
	Lacquer Spray Machine 16	3007426	190:46:47	231	16:08:08	1228	1:57:18	3200	26:15:10	133	4:46:35	0:49:33	0:04:11	65.27	93.27		263	239:53:58	0
	Lacquer Spray Machine 17	3058233	193:37:04	290	**20:14:10**	1028	1:31:48	2620	19:54:11	128	4:36:42	0:40:04	0:04:11	66.37	91.56		263	239:53:55	0
	Inside Bake Oven		239:33:10	2	0:03:02					32	0:15:45	119:46:35	0:01:31		99.98			239:51:57	0
	Inspection Camera	20872970	171:45:36	42	0:22:17			19117	67:33:24	6	0:13:50	4:05:22	0:00:32	80.53	99.85		2027	239:55:07	23
	Palletizer 1A	0		0	**239:51:54**									**0**	**0**			239:51:54	0
	Palletizer 1B	21040416	213:29:19	1205	**23:53:40**			41	2:30:59			**0:10:38**	0:01:11	**63.53**	90.04		1643	239:53:58	0

Appendix 2 Model Parameters

	Process Time (s)	Capacity	Number of	MTBF	MTTR
Cupper	0.2857	9	1	Random.Exponential(0.46065)	Random.Exponential(0.05241)
Jefferson	0.3000	1	10	Random.Exponential(0.46065)	Random.Exponential(0.05241)
BodyMaker	0.2143	1	4	Random.Exponential(1.50044)	Random.Exponential(0.15916)
	0.2804	1	2		
	0.2400	1	2		
Trimmer	0.6429	3	4	Random.Exponential(1.70084)	Random.Exponential(0.12078)
	0.8411	3	2		
	0.7200	3	2		
Washer	60.0000	1750	1	Random.Exponential(25.50981)	Random.Exponential(0.13444)
Flanger	0.6667	10	2	Random.Exponential(0.4638)	Random.Exponential(0.02333)
Beader	0.8	12	2	Random.Exponential(0.54185)	Random.Exponential(0.04102)
LightTester	0.6667	10	2	Random.Exponential(1.09259)	Random.Exponential(0.07986)
LSM	2	20	8	Random.Exponential(0.62374)	Random.Exponential(0.08017)
Paletiser	60	1800	1	Random.Exponential(0.20722)	Random.Exponential(0.02898)

Appendix 3
Custines Specification document

Author: Simon Rollinson

Date of release: 18/02/2018

Release no: 1

Authority:

Introduction to the problem

The *manufacturing beverage cans from DE-Reeler to Output of Visual Systems* process will be modelled. The objectives of the project are detailed in the next section and a wider aim of this project is to facilitate better understanding by Crown Bevcan Custines, of the process, requirements and benefits of Discrete Event Simulation.

Project objectives

The following objectives have been agreed:

1. Model the system as is and validate against appropriate production data.
2. Increasing the size of BIDI3 by up to 3 metres.
3. Effect of adding an extra canmaker.
4. Effect of adding a buffer between the Cupper and canmakers.
5.

Expected Benefits

- A better understanding of the *Production Process* by *Crown Technology*
- A better understanding of production line controls system and the potential to make improvements
- A better understanding of DES projects by Crown Beverage *Custines*
- Quantification of the results of various changes that can be made to the Production Process, in terms of altering the values of input factors (in isolation and in combination) and the consequent effects on model outputs.

Scope

This sub section outlines the boundaries of the modelling process.

- In order to avoid an overly complex simulation model (and therefore waste time on modelling experimental factors that give irrelevant or too limited information), the model only needs to include machine line controls and neglect conveyor line controls
- To avoid long simulation time, the model will be run for a
    - Maximum of 30 days of production process time and results will be extrapolated from this.
    - Each model entity will represent 10 cans for small capacity machines.
    - Each model entity will represent 50 -100 cans for large process capacity machines and conveyors.

Assumptions

The following details are assumed within the modelling process:

1. The de-reeler never runs out of stock.
2. The palletisers have infinite capacity.
3. The conveyors do not create hold ups.
4. Scrap rates are not significant
5.

General details

- Machine Operation: Cycle times for each machine have been modelled using the data provided see appendix 2.

- Machine breakdown/stoppage durations: (MTTR – Mean Time To Repair) and the frequency (MTBF – Mean Time Between Failure) have been modelled using the data from SMARTLINE *Failure Data*. Each machine dataset has been treated as an *exponential* distribution. Outlier data points within the data for each machine cycle time are included/excluded within the distributions, as these are/not considered significant events.

- Scrap rates: for each machine have been modelled using the data provided from *Spoilage Data*.

Experimental factors

- An additional canmaker machine will be added within the model. The effects on model outputs will be recorded.

- An additional capacity will be added to the BIDI following the IBO will be added within the model. The effects on model outputs will be recorded.

- An additional buffer will be added between the cupper and the canmakers will be added within the model. The effects on model outputs will be recorded.

Model requirements

- The software to be used is Simio.

- Functionally, the software only needs to provide a relatively minimal level of graphical representation.

- Outputs from Simio will be assimilated within Microsoft Excel in order to convey results.

Description of Operation

The process flow/Activity Cycle Diagram of the process that is to be modelled is shown below:

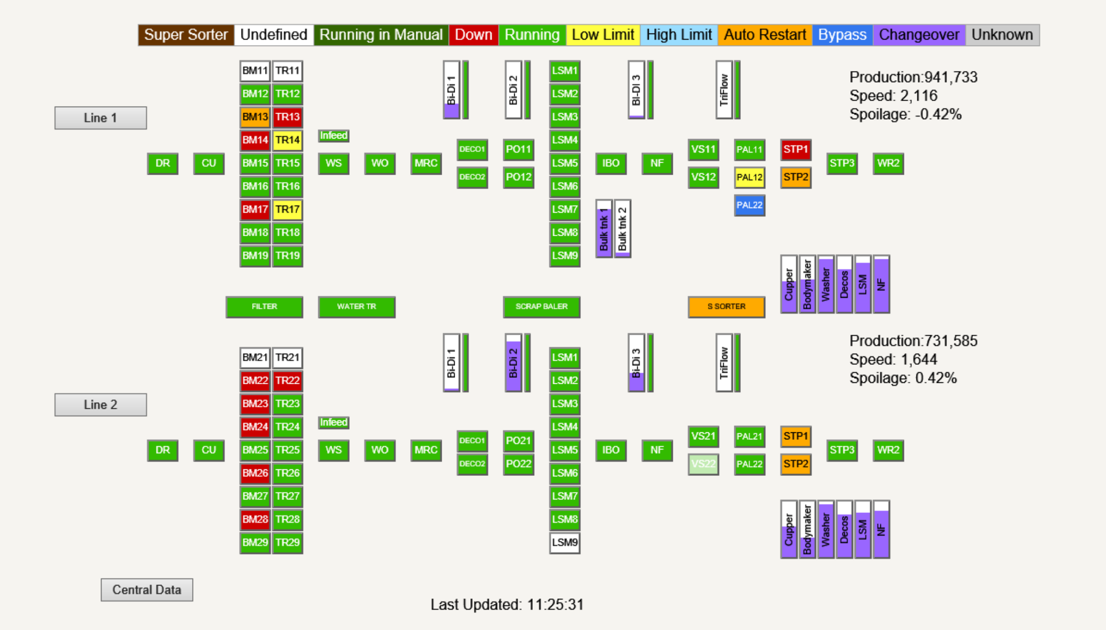


Figure 1Smartline Representation

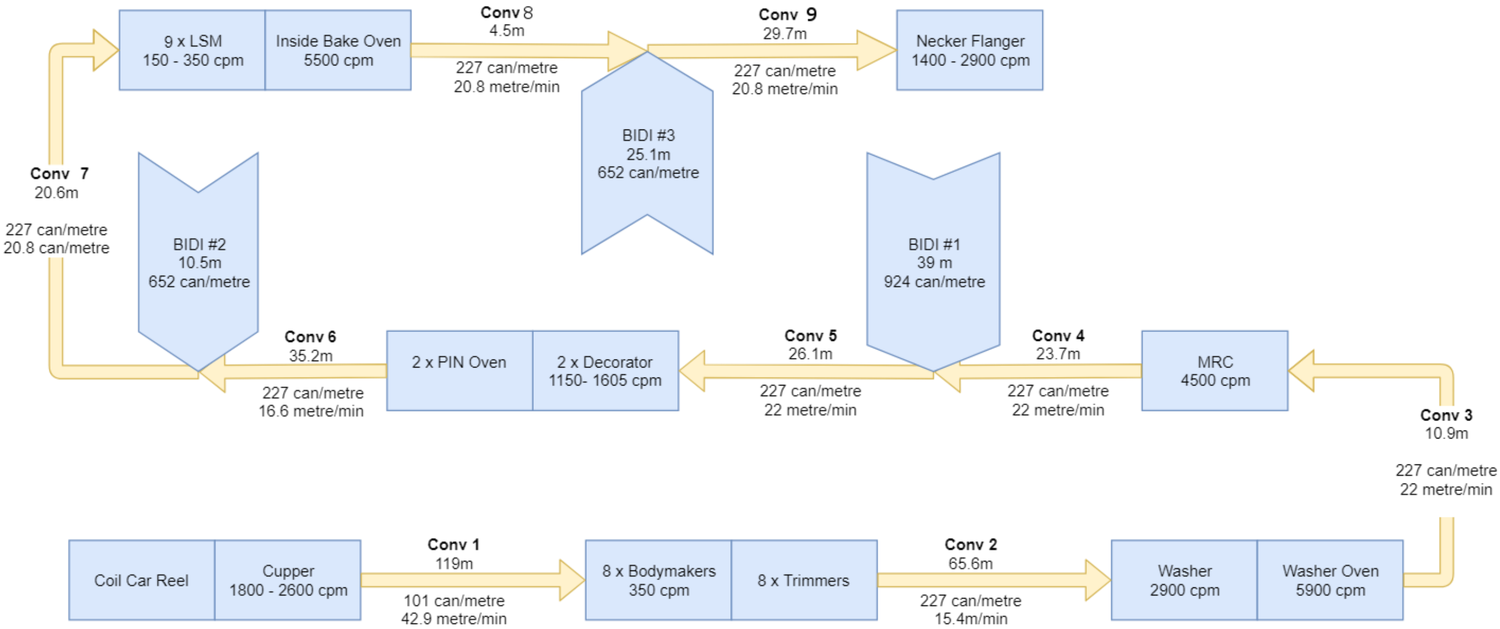


Figure 2 Outline Conceptual Model

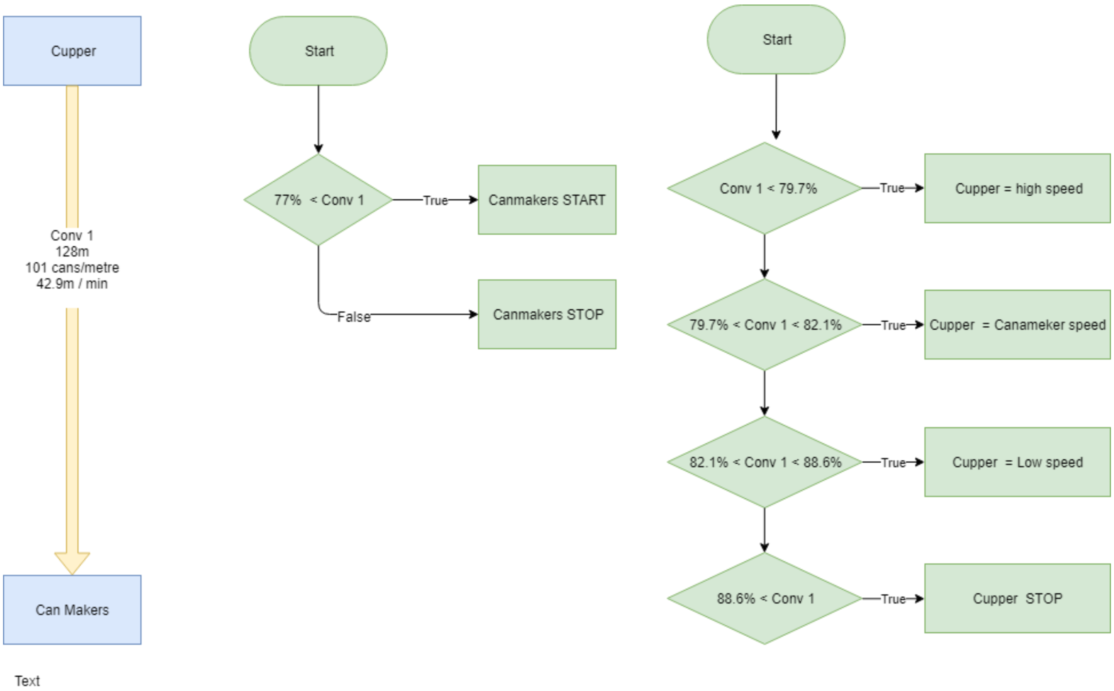


Figure 3 Cupper to Decorator Process Flow (left) and Process Logic (right).

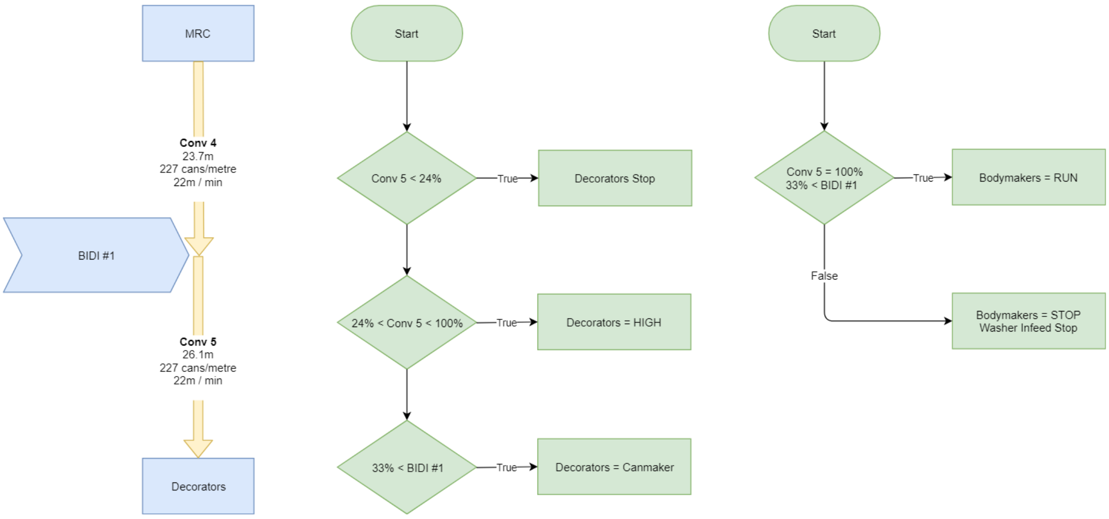


Figure 4 MRC to Decorators Process Flow (left) and Process Logic (right)

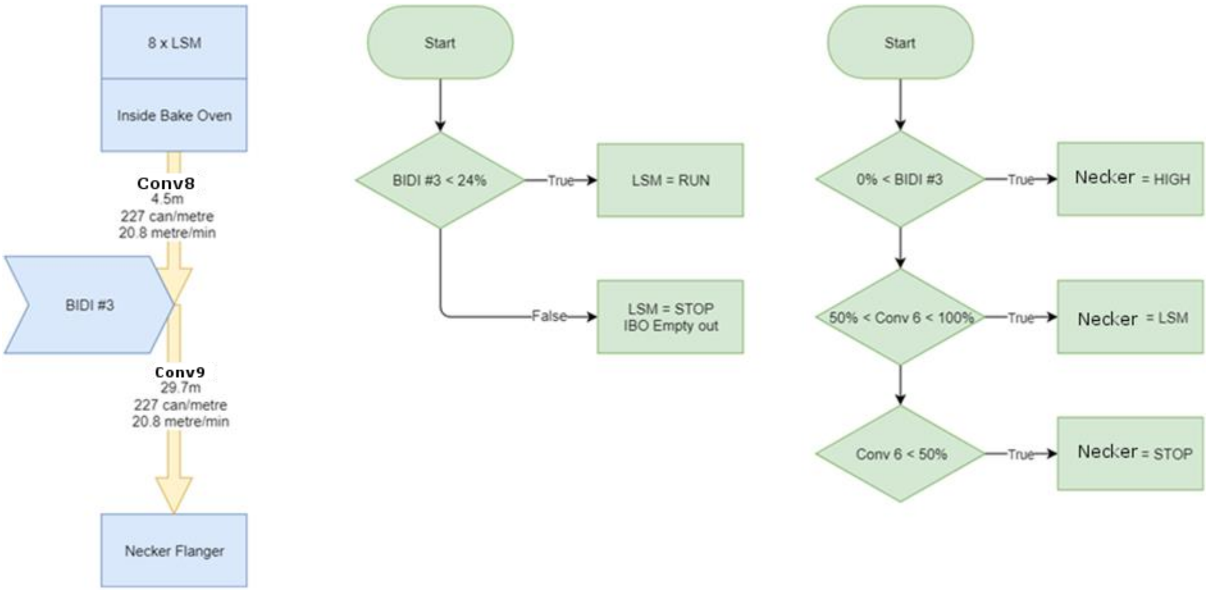


Figure 5 LSM to Necker Flanger Process Flow(left) and Process Logic (right)

Output

- Throughput of Process, under each scenario.
- Percentage machine utilisation, to include breakdown of time: idle, busy, blocked, down.
- Individual throughputs of machines on Process to achieve required bottleneck condition.
- Recommendations based on a comparison of the original performance of Process with the experimental cases.

Project Management

Simon Rollinson is the Project Manager. He will communicate directly with Crown Bevcan *Custines* as required. The formal project plan is shown below, without dates:

Task Number	Task Name	Duration	Predecessors	Resource Names
1	Visit 1 - Familiarisation	1 day		
2	Project Management	1 day		
3	Visit 2 - Data Collection	2 day	5	

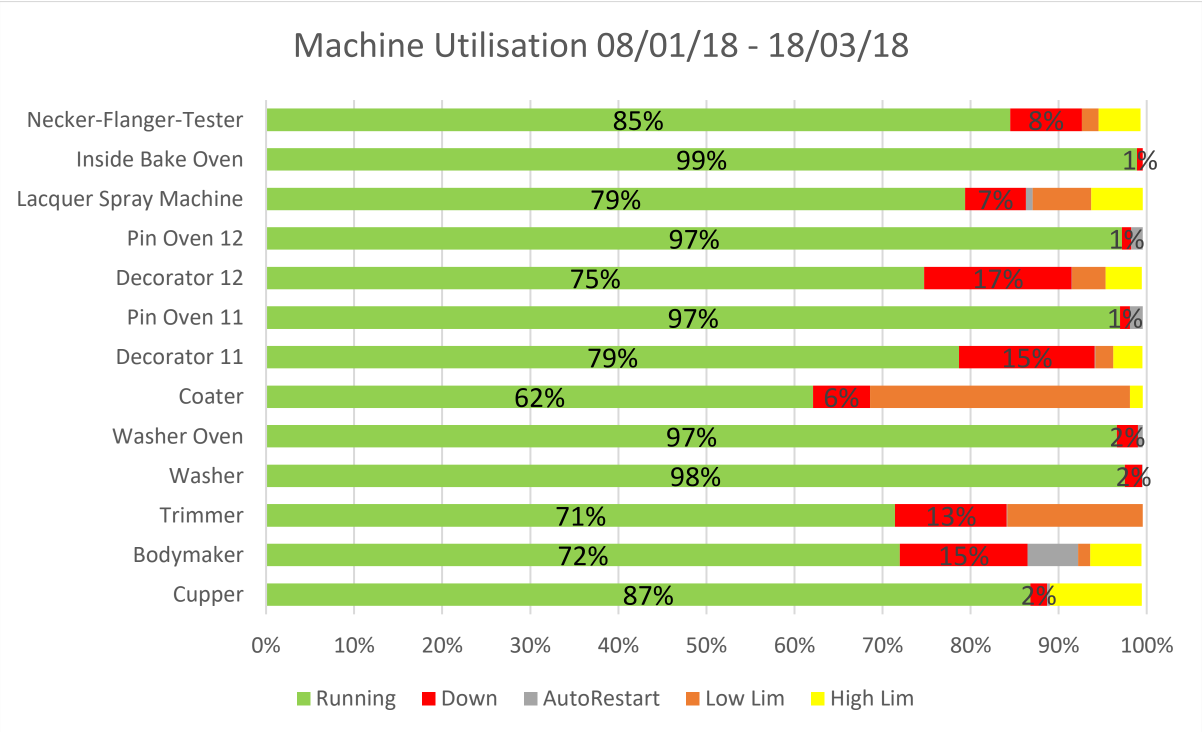
4	Conceptualisation - ACD and Process Flow Diagram	5 day	1	
5	Specification Document	3 day	3	
6	Validation 1	1 day	3	
7	DES Modelling	5 days	6	
8	Model Validation	1 day	7	
9	Experimentation	4 days	8	
10	Results	1 day	9	
11	Documentation	2 days	10	
12	Refinement/Contingency	3 days		

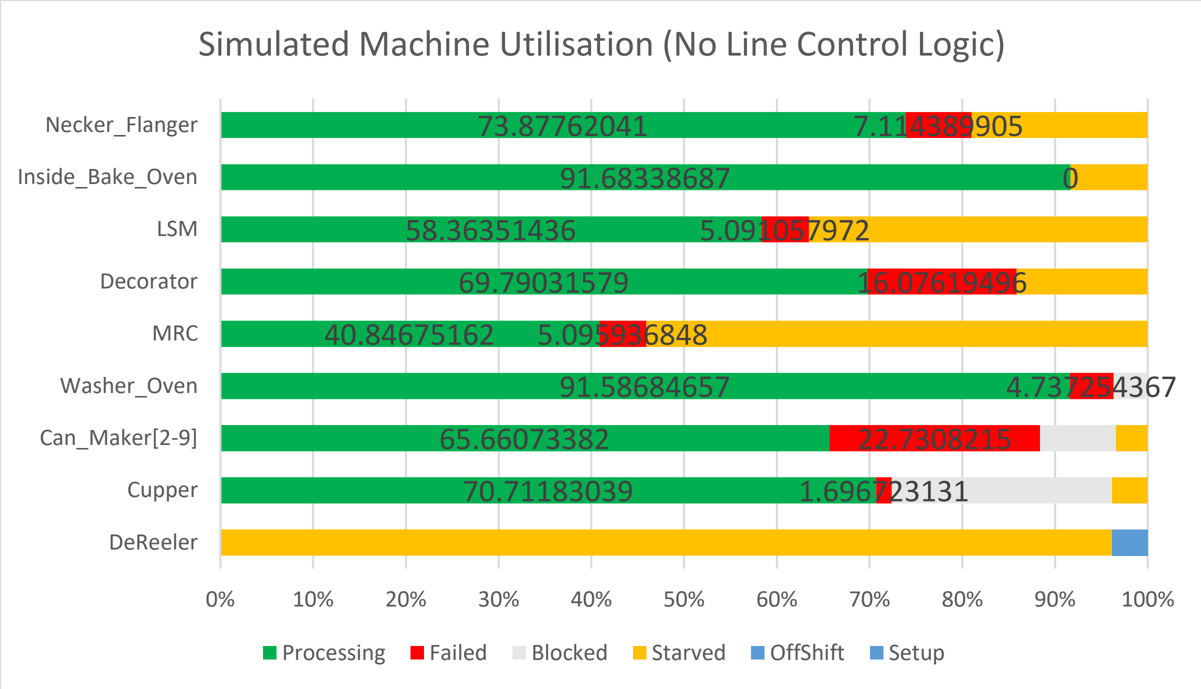
Xhrs (Research Engineer) + Yhrs (Senior Research Engineer)

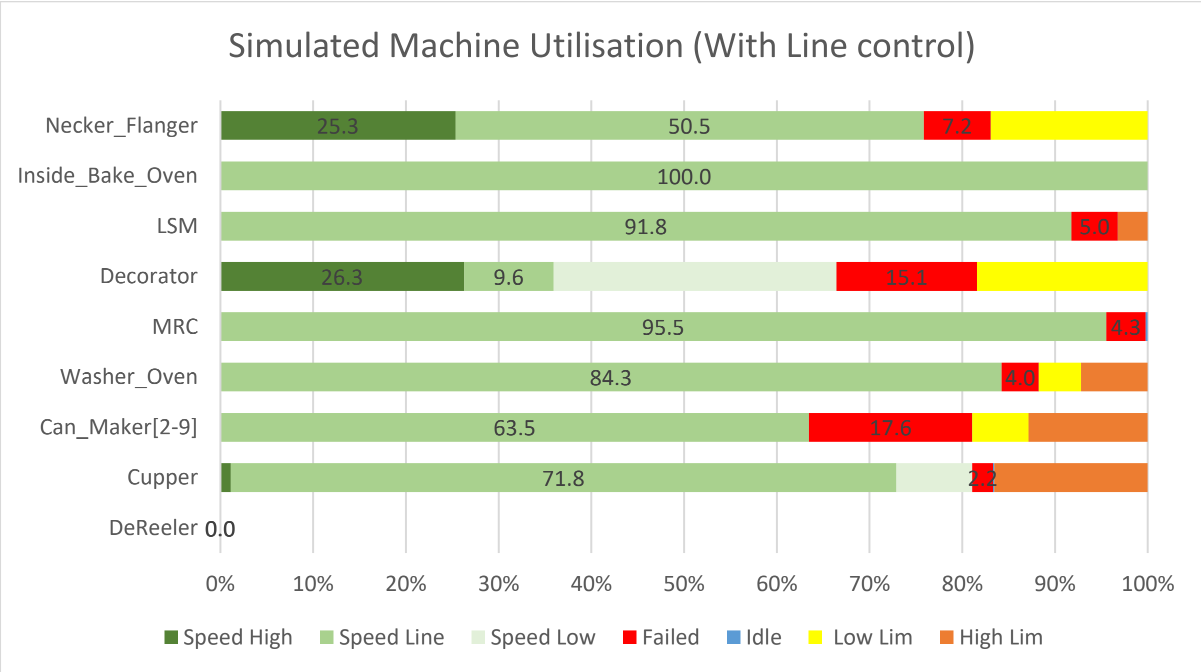
Results

1. Validation

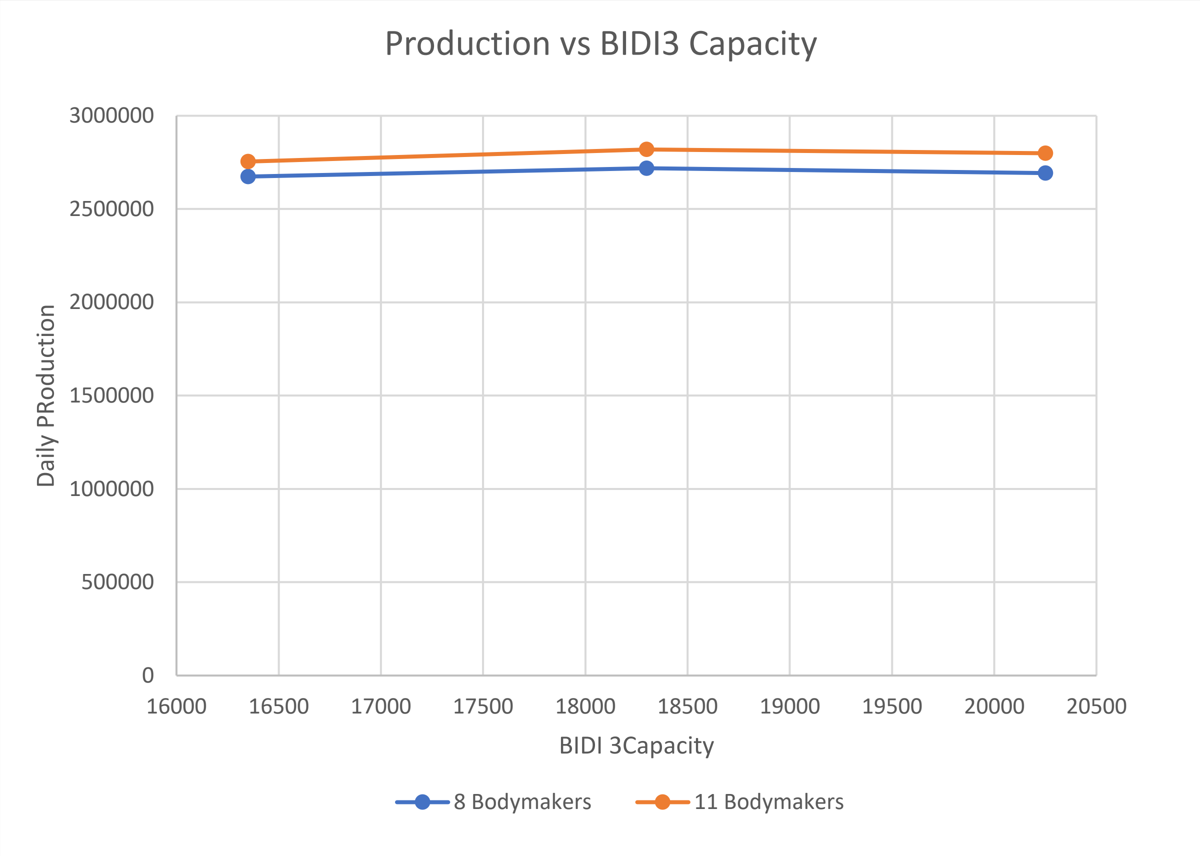
The Smartline system shows average daily production at 2.69M cans with a standard deviation of 0.28M cans for the period 08/01/18 to 18/03/18. The same period was chosen.




Simulated Machine Utilisation (No Line Control Logic)
Necker_Flanger
Inside_Bake_Oven
LSM
Decorator
MRC
Washer_Oven
Can_Maker[2-9]
Cupper
DeReeler
73.87762041
7.114389905
91.68338687
0
58.36351436
5.091057972
69.79031579
16.07619496
40.84675162
5.095936848
91.58684657
4.737254367
65.66073382
22.7308215
70.71183039
1.696723131
0% 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%
Processing Failed Blocked Starved OffShift Setup


Simulated Machine Utilisation (With Line control)
Necker_Flanger
Inside_Bake_Oven
LSM
Decorator
MRC
Washer_Oven
Can_Maker[2-9]
Cupper
DeReeler
25.3
50.5
7.2
100.0
91.8
5.0
26.3
9.6
15.1
95.5
4.3
84.3
4.0
63.5
17.6
71.8
2.2
0.0
0% 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%
Speed High Speed Line Speed Low Failed Idle Low Lim High Lim

2. BIDI 3 Capacity



3. Canmaker Study

- The model was run with Jan18 and Feb 18 data.
- 5 days with 0.25 day warmup period; 40 replications for each run.

At a contribution of $15 per 1000 cans, the payback period for the 9th Bodymaker is between 215 and 247 days.

The payback for the 10th Bodymaker is not guaranteed.

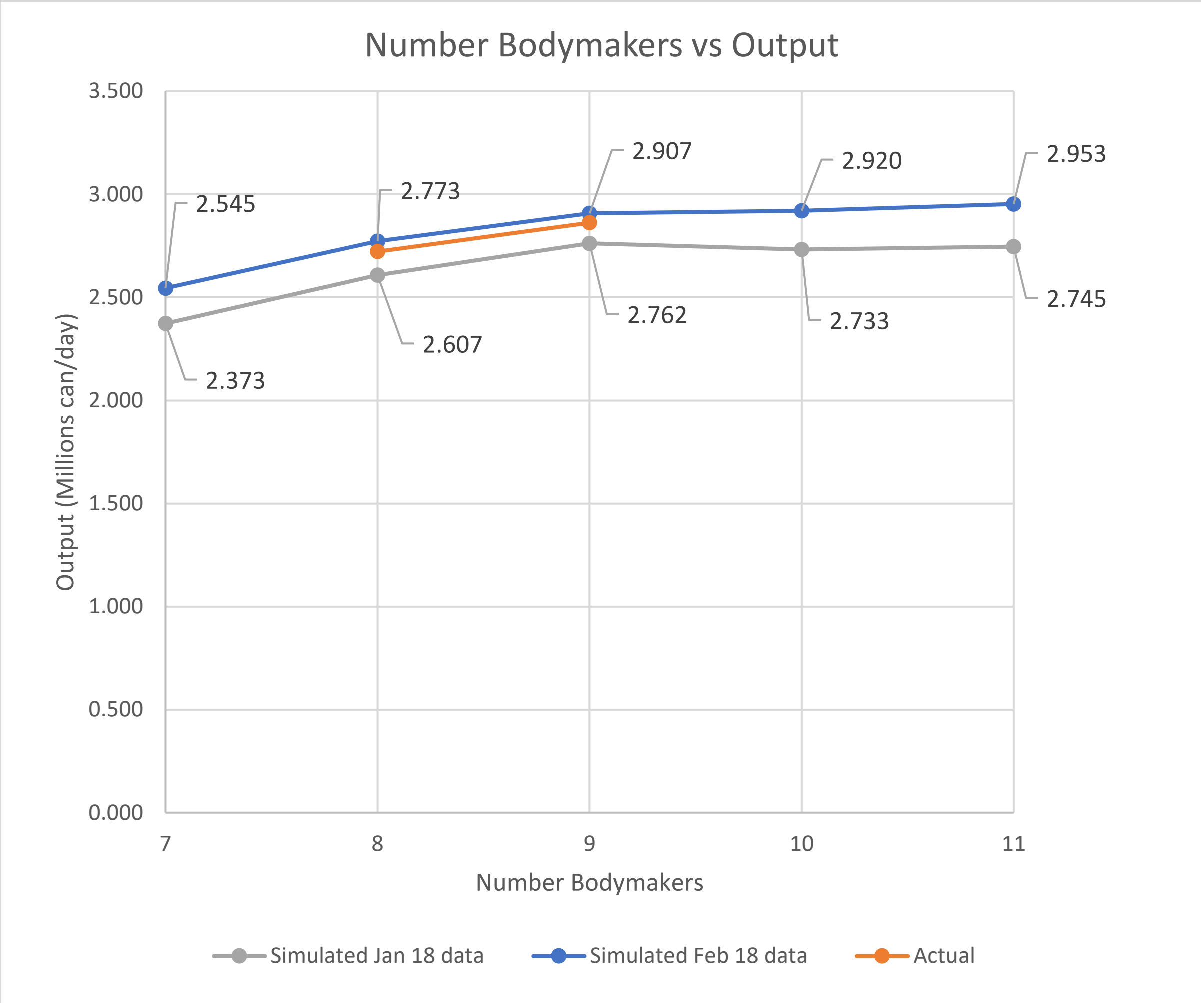
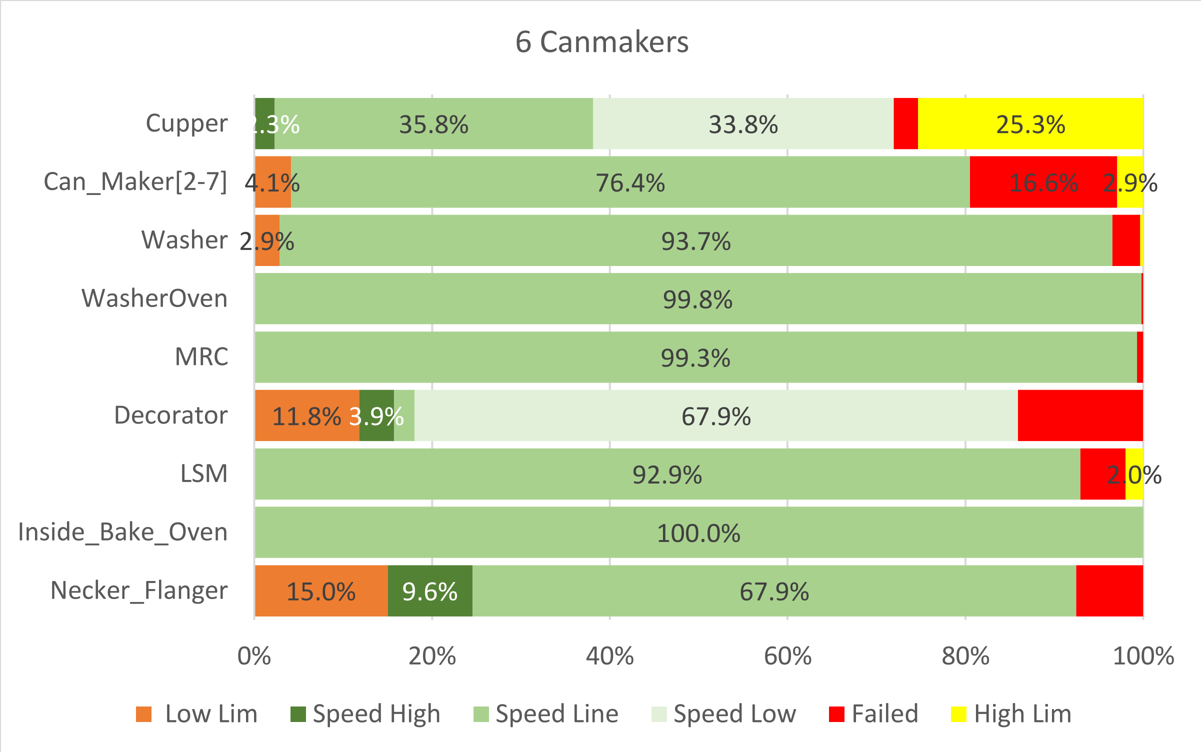


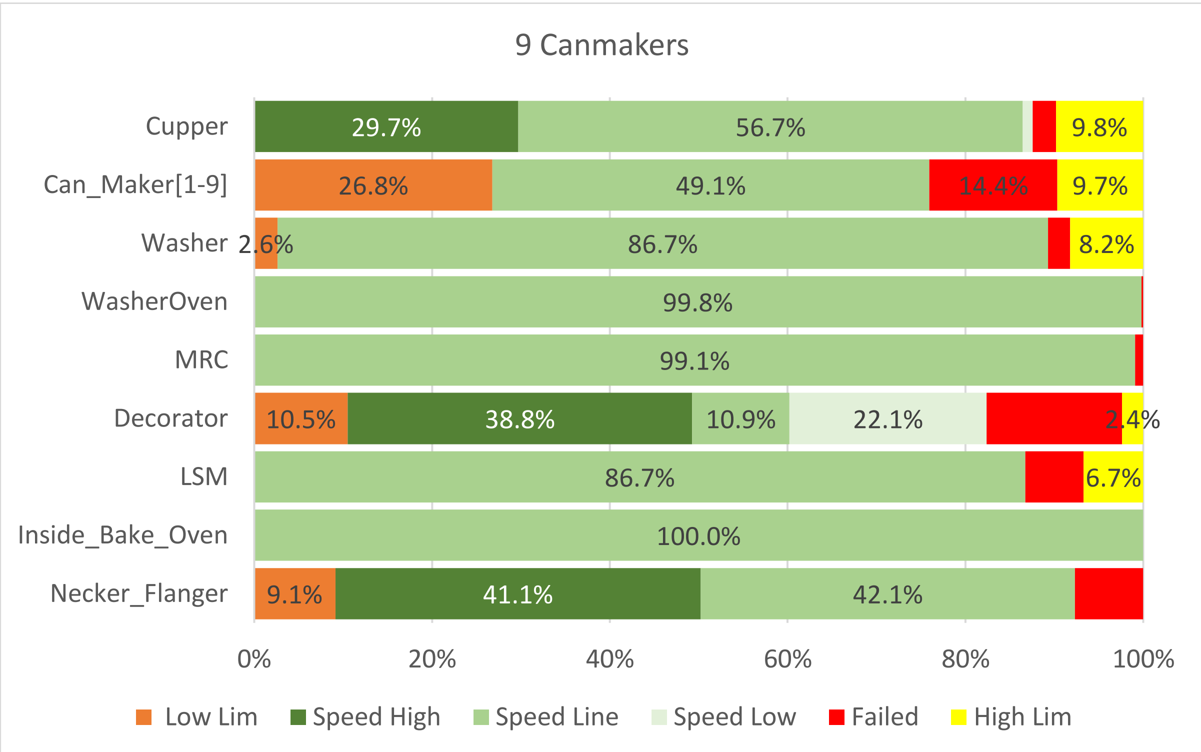
Figure 6 Custines Effect of Number of Canmakers on Output

The number of canmakers is optimal at 9 canmakers. The data suggest that based upon eight canmakers, nine canmakers would increase production by 4.7%. Ten canmakers would increase production by 6.2% (an extra 1.5%). Eleven canmakers yields only an extra 0.5%.

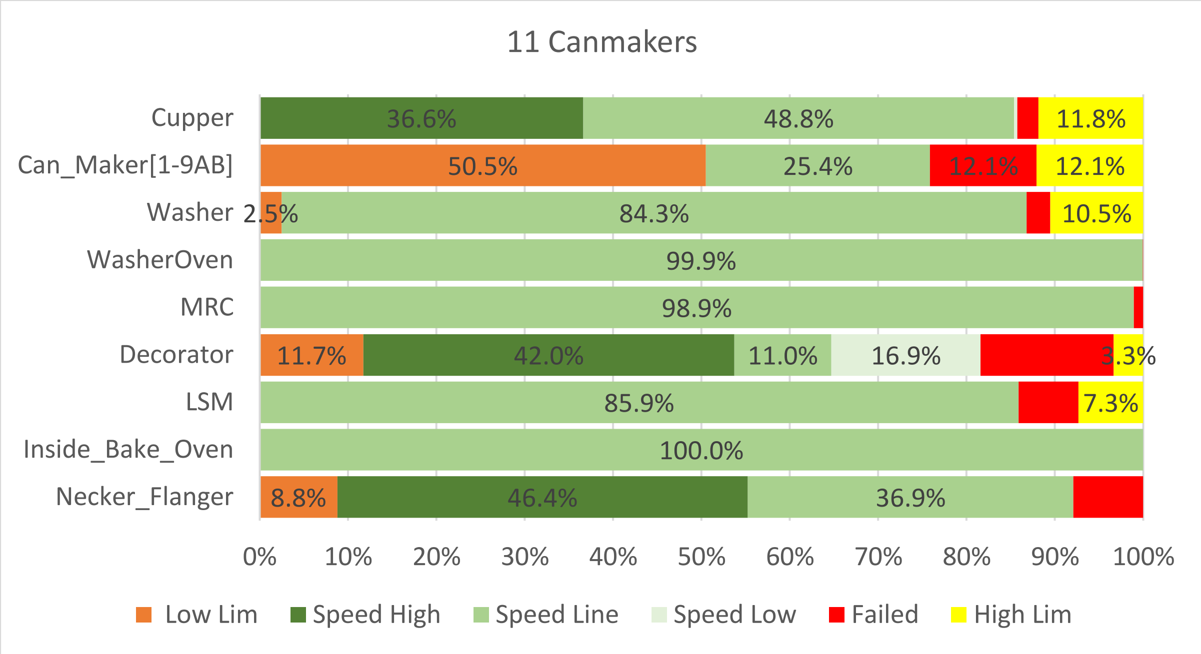
Assuming $15 per 1000 cans contribution and a cost of a Bodymaker approximately $500,000, then the payback period for the 9th Bodymaker is



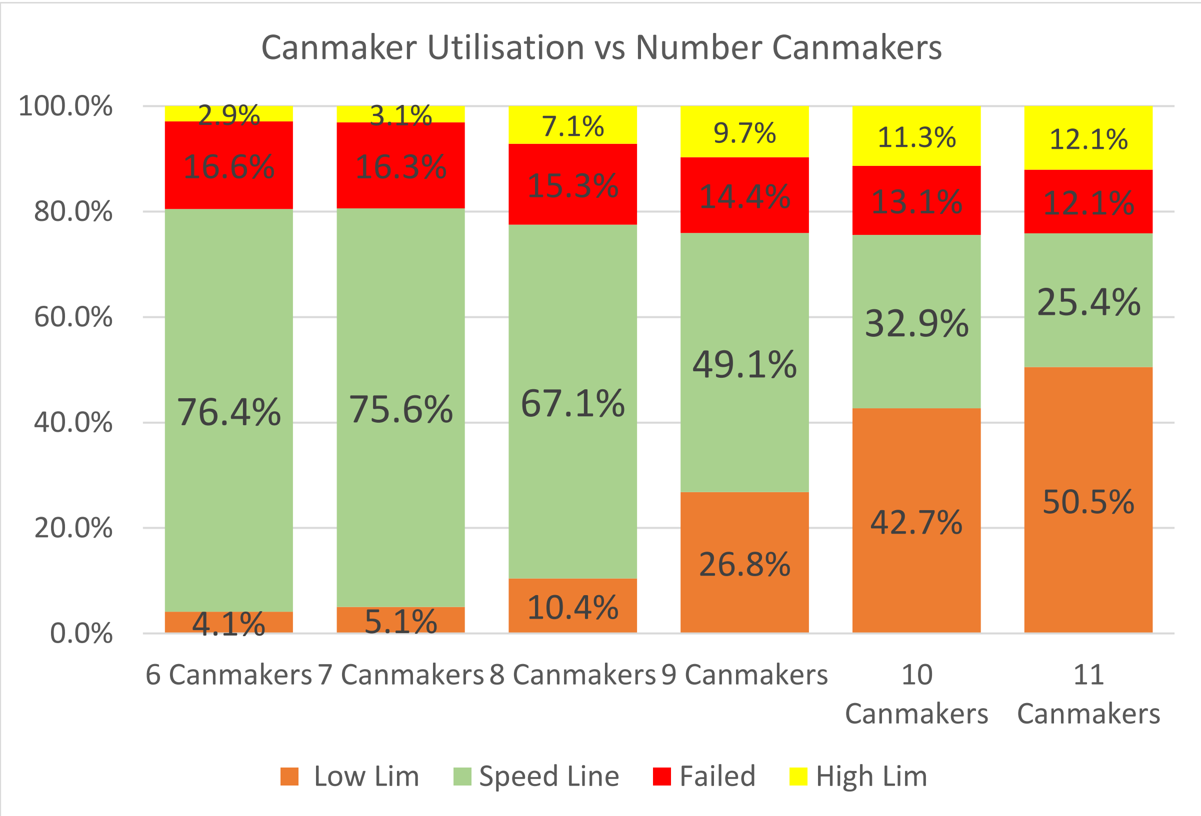
With only 6 canmakers, the cupper is underutilised and spends most of its time running at low speed and is blocked for almost half the time



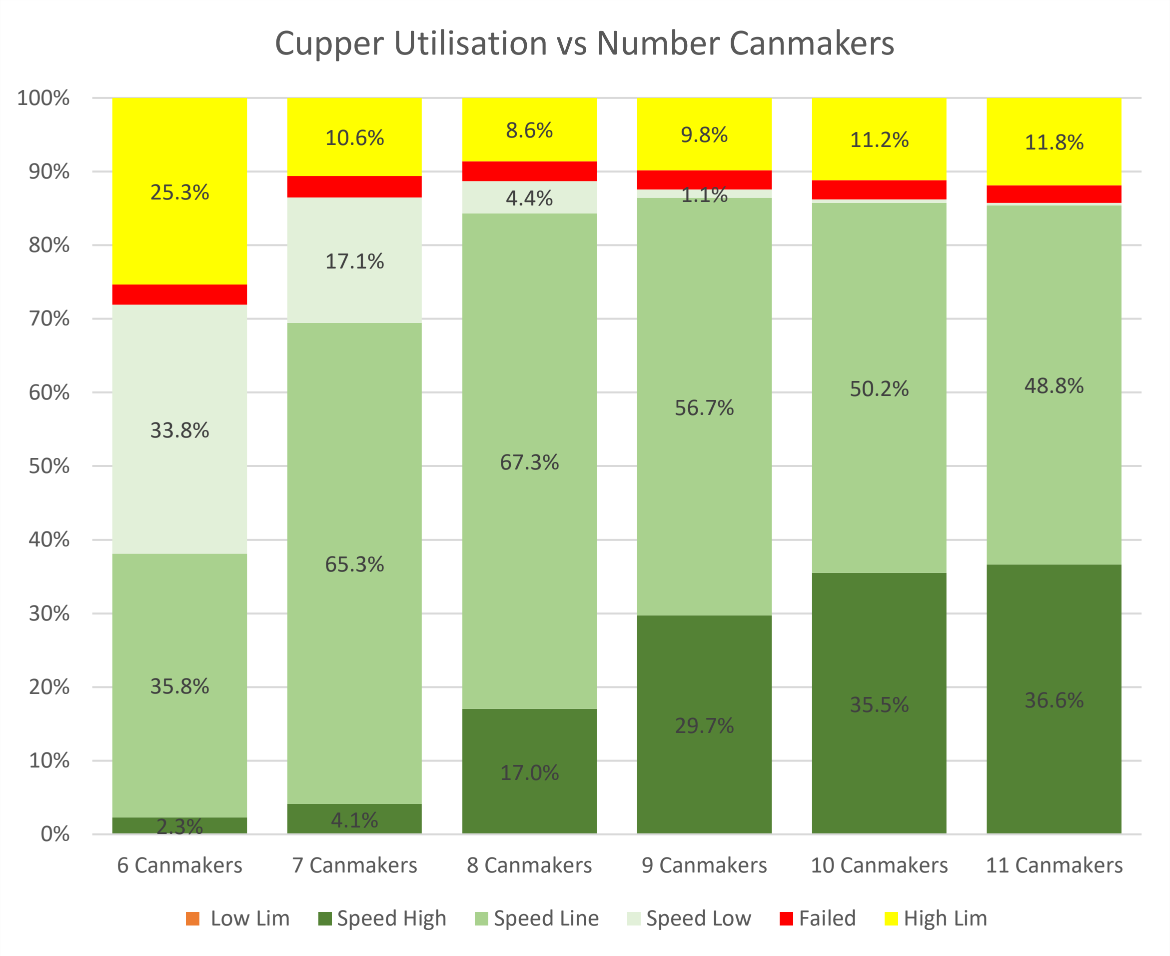
With 9 canmakers, the cupper is able to spend much more time running at the higher speed but they are blocked for more time from 3% to 10%.



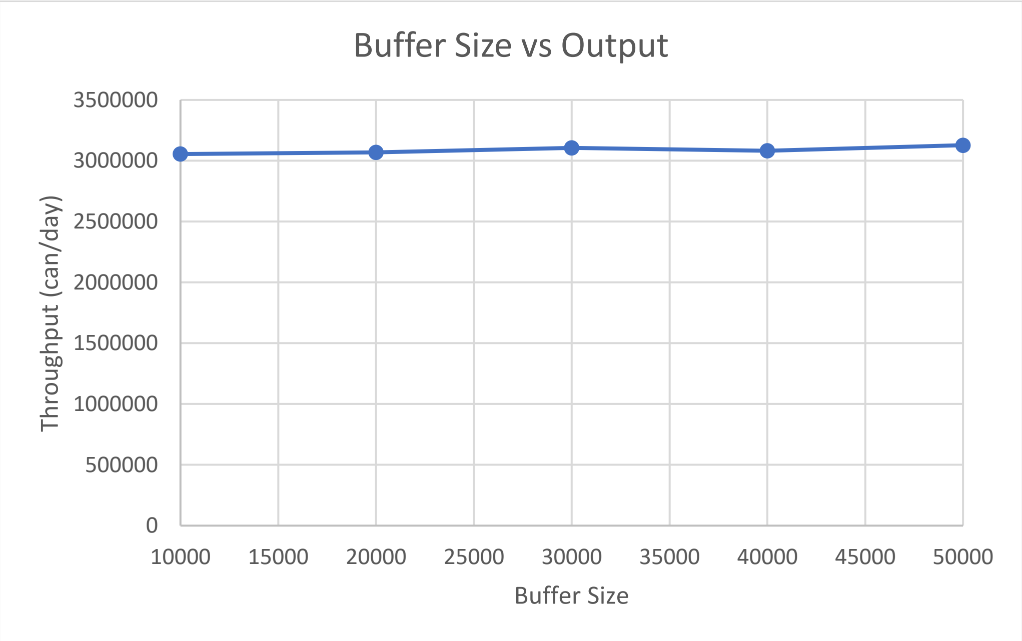
With 11 canmakers, the cupper can spend a little more time running fast, but now the canmakers are becoming starved for 50% of the time.

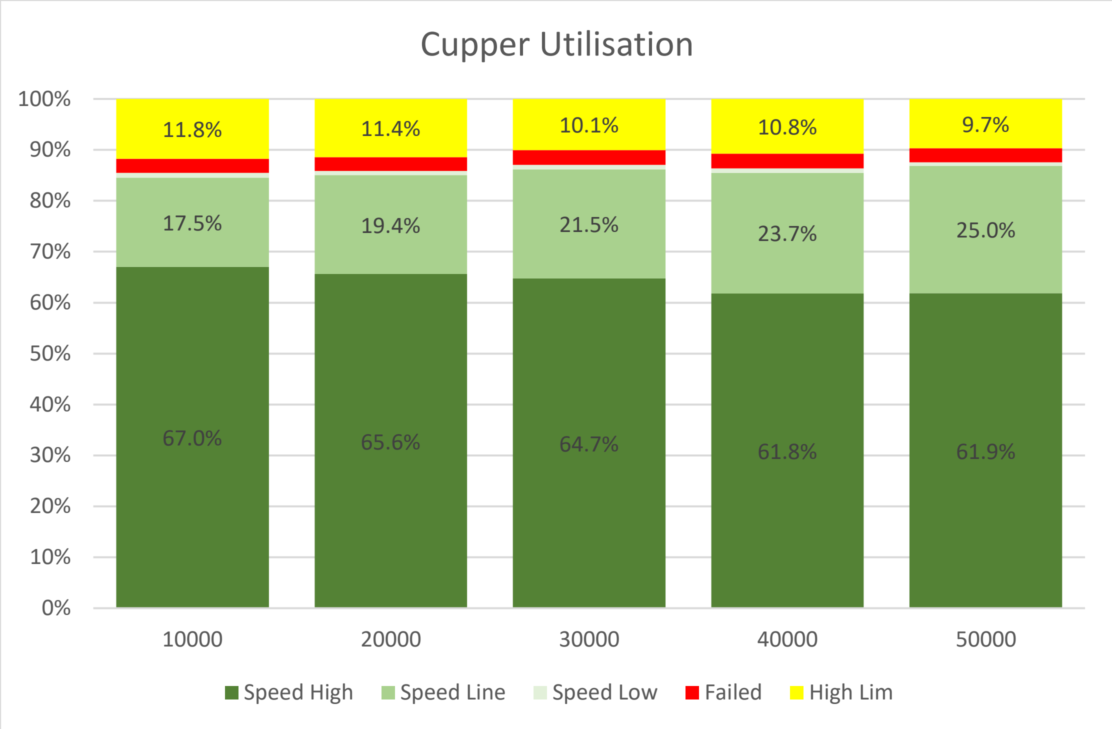


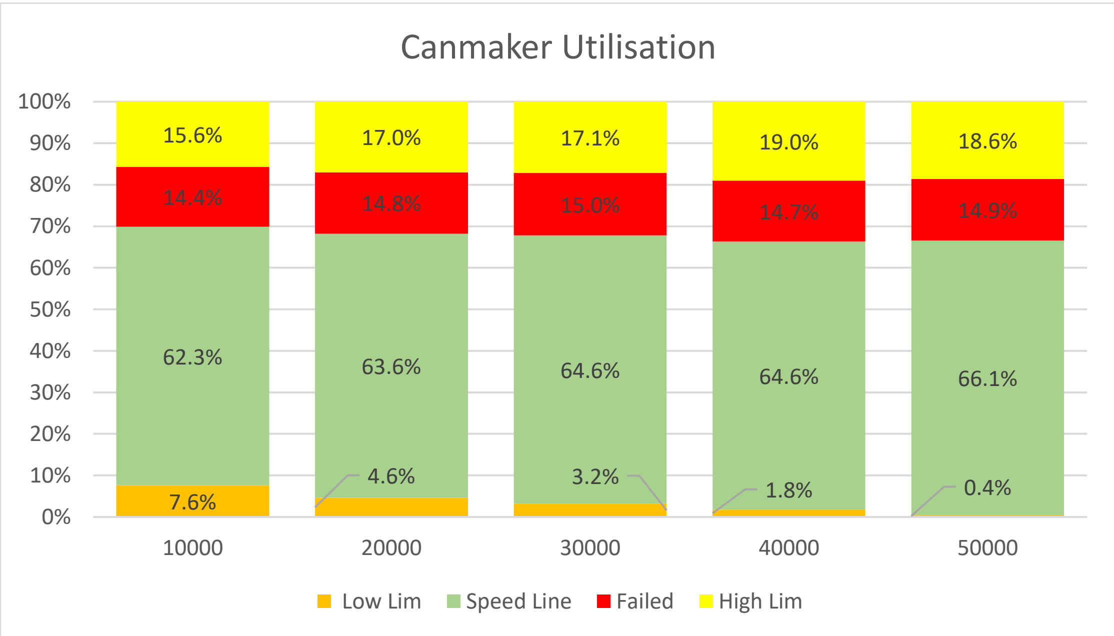
The canmaker utilisation begins to drop off significantly at 10 canmakers. 9 Canmakers is optimal for this line.



4. Buffer between Cupper & Canmaker




Cupper Utilisation
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
11.8%
11.4%
10.1%
10.8%
9.7%
17.5%
19.4%
21.5%
23.7%
25.0%
67.0%
65.6%
64.7%
61.8%
61.9%
10000
20000
30000
40000
50000
Speed High
Speed Line
Speed Low
Failed
High Lim


Canmaker Utilisation
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
15.6%
17.0%
17.1%
19.0%
18.6%
14.4%
14.8%
15.0%
14.7%
14.9%
62.3%
63.6%
64.6%
64.6%
66.1%
7.6%
4.6%
3.2%
1.8%
0.4%
10000
20000
30000
40000
50000
Low Lim
Speed Line
Failed
High Lim

Conclusions

1. The baseline model provides a good match for the physical system.
2. Increasing the capacity of the BiDi following the IBO would not yield extra output
3. The optimum number of Bodymakers is 9.
4. An extra buffer between the Cupper and Bodymaker group would not yield extra output.

Appendix 1 – Line Control Logic

Custines Line 1

	Machine Speed Cans/Min		Conveyor Max Speed		Conveyor Distance Machine to Sensor Effect							Can Capacity			
Machine	Maximum	Minimum	Cans/Min	M/Min	Overall	Restart		Line Speed		Buildback		Per M	To Restart	To Line Speed	
							Sensor ID		Sensor ID		Sensor ID				
Cupper	3220	1680	4340	42.9	128.3m					8.5m	S110B	101		11493.8	If less than Line Speed go to Max Speed
Canmakers	2800	2000	3900	15.4	65.6m			42m	S226C	2.2m	S141C	227		21010.5	Line Speed - Deco Infeed Full +30% Washer BiDi
															If less than Line Speed All Canmakers go to Max Speed
Washer	5.3m/min = 3000cpm		5000	22	59m					39.2m	S226B	227			
BiDi			3500	3.7	45m							924			
Decorator	1605	1150				15m	S230A	42m	S226C			227	3405	21010.5	Line Speed - Deco Infeed Full +30% Washer BiDi
Decorator Leg 1			4000	16.6	22m					12m	S305B	227			
Decorator Leg 2			4000	16.6	25m					13m	S318B	227			
Decorator Combined			5000	20.8	30m							227			
BiDi			2500	2.6	10.5m							652			
LSM/IBO	1200	2450	5000	20.8	26m	11m	S339A	18.5m	S337A	25m	S405D	227	2497	4199.5	
Bidi			2600	4.1	25m							682			
Necker	1400	2500	4000	15	36m	13m	S409A	24m	S407B	17m	S456A	227	2951	5448	
Palletiser Area	4200		4000	12.5	74m	4.5m	S489A					303	1363.5		

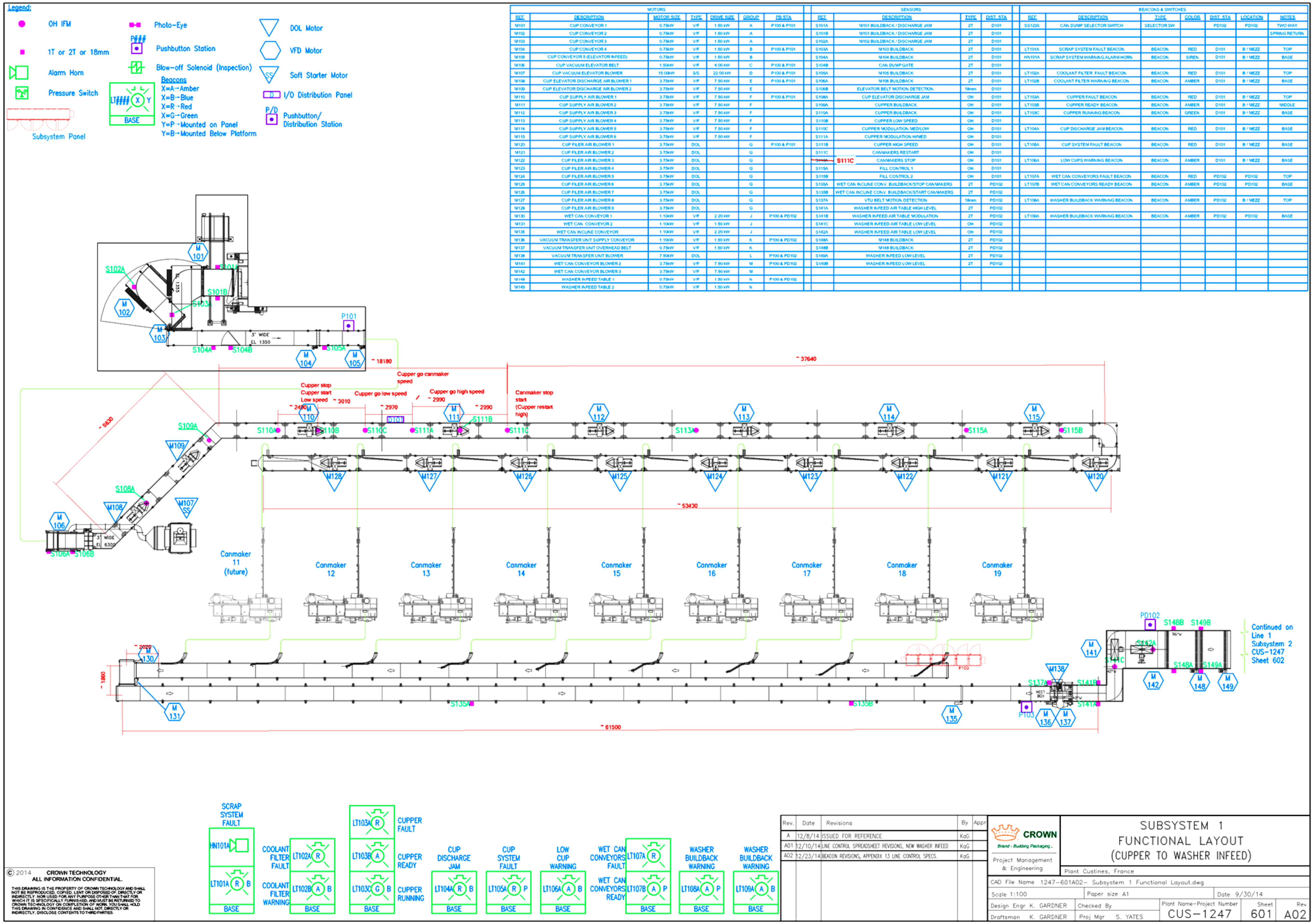
Buildback Sensor is downstream conveying

Appendix 2: Process Times

No	Process Name	Processing Time						O.E.E. (%) (if available or combination of uptime and throughput)	Breakdown data		Reject Rate/Scrap Rate	Specific Questions	Answer
		Setup Time	Processing Time can/min		Machine Capacity (Single/parrallel processesing)	Processing Time seconds/can			MTTR	MTBF			
			Max	Min		Max	Min						
1	De Reeler		150	300	1	0.4000	0.2000				?	How many cups per reel ? Range of times for reel changeover ?	
2	Cupper		1800	2600	1	0.0333	0.0231				?		
3	Body Maker		350	350	1	0.1714	0.1714				?		
4	Trimmer		350	350	1	0.1714	0.1714				?	Can we use the Body maker process time ?	
5	Washer & Oven		2900	3000	3000	62.0690	60.0000				?	approx 1 minute processing time ?	
6	MRC		4500	4500	1	0.0133	0.0133				?		
7	Decorator PIN Oven		1150	1605	250	13.0435	9.3458				?	approx 10 seconds processing time ?	
8	LSM		150	350	1	0.4000	0.1714				?		
9	Inside Bake Oven		5500	5500	5500	60.0000	60.0000				?	approx 1 minute processing time ?	
10	Necker Flanger		1400	2800	12	0.5143	0.2571				?	12 head machine ?	
11													

SR to collect from Smartline

Appendix 3 – CAD Layout



MOTORS

REF	DESCRIPTION	MOTOR SIZE	TYPE	DRIVE SIZE	GROUP	PB STA
M101	CUP CONVEYOR 1	0.75kW	V/F	1.50 kW	A	P100 & P101
M102	CUP CONVEYOR 2	0.75kW	V/F	1.50 kW	A	
M103	CUP CONVEYOR 3	0.75kW	V/F	1.50 kW	A	
M104	CUP CONVEYOR 4	0.75kW	V/F	1.50 kW	B	P100 & P101
M105	CUP CONVEYOR 5 (ELEVATOR INFEED)	0.75kW	V/F	1.50 kW	B	
M106	CUP VACUUM ELEVATOR BELT	1.50kW	V/F	4.00 kW	C	P100 & P101
M107	CUP VACUUM ELEVATOR BLOWER	15.00kW	S/S	22.00 kW	D	P100 & P101
M108	CUP ELEVATOR DISCHARGE AIR BLOWER 1	3.75kW	V/F	7.50 kW	E	P100 & P101
M109	CUP ELEVATOR DISCHARGE AIR BLOWER 2	3.75kW	V/F	7.50 kW	E	
M110	CUP SUPPLY AIR BLOWER 1	3.75kW	V/F	7.50 kW	F	P100 & P101
M111	CUP SUPPLY AIR BLOWER 2	3.75kW	V/F	7.50 kW	F	
M112	CUP SUPPLY AIR BLOWER 3	3.75kW	V/F	7.50 kW	F	
M113	CUP SUPPLY AIR BLOWER 4	3.75kW	V/F	7.50 kW	F	
M114	CUP SUPPLY AIR BLOWER 5	3.75kW	V/F	7.50 kW	F	
M115	CUP SUPPLY AIR BLOWER 6	3.75kW	V/F	7.50 kW	F	
M120	CUP FILER AIR BLOWER 1	3.75kW	DOL		G	P100 & P101
M121	CUP FILER AIR BLOWER 2	3.75kW	DOL		G	
M122	CUP FILER AIR BLOWER 3	3.75kW	DOL		G	
M123	CUP FILER AIR BLOWER 4	3.75kW	DOL		G	
M124	CUP FILER AIR BLOWER 5	3.75kW	DOL		G	
M125	CUP FILER AIR BLOWER 6	3.75kW	DOL		G	
M126	CUP FILER AIR BLOWER 7	3.75kW	DOL		G	
M127	CUP FILER AIR BLOWER 8	3.75kW	DOL		G	
M128	CUP FILER AIR BLOWER 9	3.75kW	DOL		G	
M130	WET CAN CONVEYOR 1	1.10kW	V/F	2.20 kW	J	P100 & PD102
M131	WET CAN CONVEYOR 2	1.10kW	V/F	1.50 kW	J	
M135	WET CAN INCLINE CONVEYOR	1.10kW	V/F	2.20 kW	J	
M136	VACUUM TRANSFER UNIT SUPPLY CONVEYOR	1.10kW	V/F	1.50 kW	K	P100 & PD102
M137	VACUUM TRANSFER UNIT OVERHEAD BELT	0.75kW	V/F	1.50 kW	K	
M138	VACUUM TRANSFER UNIT BLOWER	7.50kW	DOL		L	P100 & PD102
M141	WET CAN CONVEYOR BLOWER 2	3.75kW	V/F	7.50 kW	M	P100 & PD102
M142	WET CAN CONVEYOR BLOWER 3	3.75kW	V/F	7.50 kW	M	
M148	WASHER INFEED TABLE 1	0.75kW	V/F	1.50 kW	N	P100 & PD102
M149	WASHER INFEED TABLE 2	0.75kW	V/F	1.50 kW	N	

SENSORS

REF	DESCRIPTION	TYPE	DIST. STA
S101A	M101 BUILDBACK / DISCHARGE JAM	2T	D101
S101B	M101 BUILDBACK / DISCHARGE JAM	2T	D101
S102A	M102 BUILDBACK / DISCHARGE JAM	2T	D101
S103A	M103 BUILDBACK	2T	D101
S104A	M104 BUILDBACK	2T	D101
S104B	CAN DUMP GATE	2T	D101
S105A	M105 BUILDBACK	2T	D101
S106A	M106 BUILDBACK	2T	D101
S106B	ELEVATOR BELT MOTION DETECTION	18mm	D101
S108A	CUP ELEVATOR DISCHARGE JAM	OH	D101
S109A	CUPPER BUILDBACK	OH	D101
S110A	CUPPER BUILDBACK	OH	D101
S110B	CUPPER LOW SPEED	OH	D101
S110C	CUPPER MODULATION MED/LOW	OH	D101
S111A	CUPPER MODULATION HI/MED	OH	D101
S111B	CUPPER HIGH SPEED	OH	D101
S111C	CANMAKERS RESTART	OH	D101
~~S113A~~ S111C	CANMAKERS STOP	OH	D101
S115A	FILL CONTROL 1	OH	D101
S115B	FILL CONTROL 2	OH	D101
S135A	WET CAN INCLINE CONV. BUILDBACK/STOP CANMAKERS	2T	PD102
S135B	WET CAN INCLINE CONV. BUILDBACK/START CANMAKERS	2T	PD102
S137A	VTU BELT MOTION DETECTION	18mm	PD102
S141A	WASHER INFEED AIR TABLE HIGH LEVEL	2T	PD102
S141B	WASHER INFEED AIR TABLE MODULATION	2T	PD102
S141C	WASHER INFEED AIR TABLE LOW LEVEL	OH	PD102
S142A	WASHER INFEED AIR TABLE LOW LEVEL	OH	PD102
S148A	M148 BUILDBACK	2T	PD102
S148B	M148 BUILDBACK	2T	PD102
S149A	WASHER INFEED LOW LEVEL	2T	PD102
S149B	WASHER INFEED LOW LEVEL	2T	PD102

BEACONS & SWITCHES

REF	DESCRIPTION	TYPE	COLOR	DIST. STA	LOCATION	NOTES
SS123X	CAN DUMP SELECTOR SWITCH	SELECTOR SW		PD102	PD102	TWO WAY
						SPRING RETURN
LT101A	SCRAP SYSTEM FAULT BEACON	BEACON	RED	D101	B / MEZZ	TOP
HN101A	SCRAP SYSTEM WARNING ALARM HORN	BEACON	SIREN	D101	B / MEZZ	BASE
LT102A	COOLANT FILTER FAULT BEACON	BEACON	RED	D101	B / MEZZ	TOP
LT102B	COOLANT FILTER WARNING BEACON	BEACON	AMBER	D101	B / MEZZ	BASE
LT103A	CUPPER FAULT BEACON	BEACON	RED	D101	B / MEZZ	TOP
LT103B	CUPPER READY BEACON	BEACON	AMBER	D101	B / MEZZ	MIDDLE
LT103C	CUPPER RUNNING BEACON	BEACON	GREEN	D101	B / MEZZ	BASE
LT104A	CUP DISCHARGE JAM BEACON	BEACON	RED	D101	B / MEZZ	BASE
LT105A	CUP SYSTEM FAULT BEACON	BEACON	RED	D101	B / MEZZ	BASE
LT106A	LOW CUPS WARNING BEACON	BEACON	AMBER	D101	B / MEZZ	BASE
LT107A	WET CAN CONVEYORS FAULT BEACON	BEACON	RED	PD102	PD102	TOP
LT107B	WET CAN CONVEYORS READY BEACON	BEACON	AMBER	PD102	PD102	BASE
LT108A	WASHER BUILDBACK WARNING BEACON	BEACON	AMBER	PD102	B / MEZZ	TOP
LT109A	WASHER BUILDBACK WARNING BEACON	BEACON	AMBER	PD102	PD102	BASE

Rev.	Date	Revisions	By	Appr
A	12/8/14	ISSUED FOR REFERENCE	KaG	
A01	12/10/14	LINE CONTROL SPREADSHEET REVISIONS, NEW WASHER INFEED	KaG	
A02	12/23/14	BEACON REVISIONS, APPENDIX 13 LINE CONTROL SPECS.	KaG	

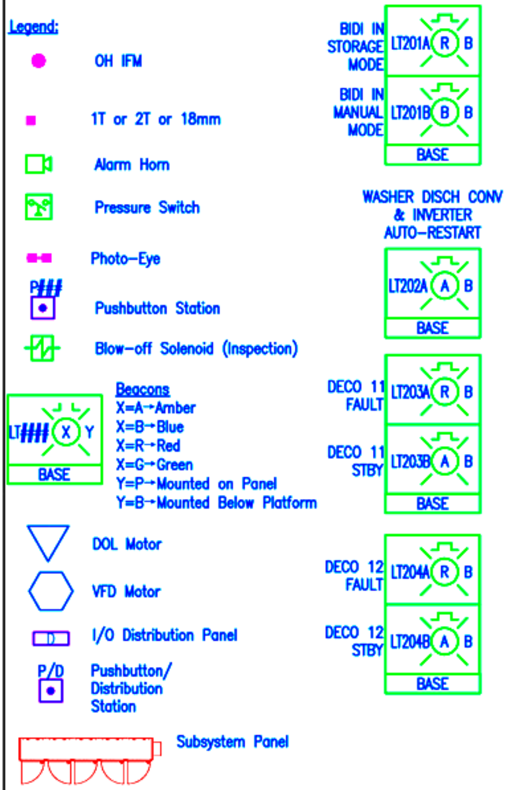
CROWN Brand-Building Packaging
Project Management & Engineering

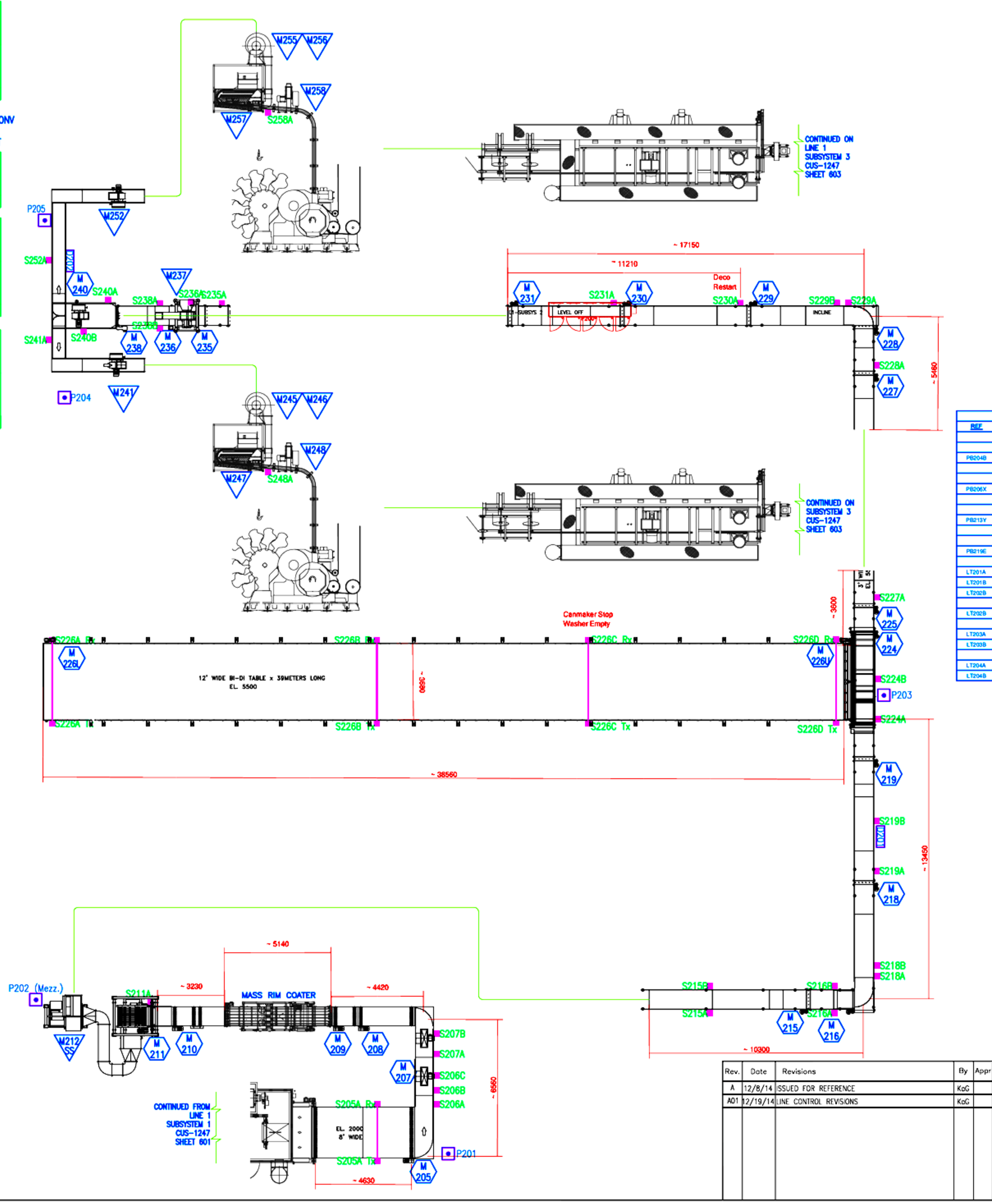
SUBSYSTEM 1
FUNCTIONAL LAYOUT
(CUPPER TO WASHER INFEED)

Plant Custines, France
CAD File Name 1247-601A02- Subsystem 1 Functional Layout.dwg
Scale 1:100 | Paper size A1 | Date 9/30/14
Design Engr K. GARDNER | Checked By | Plant Name-Project Number CUS-1247 | Sheet 601 | Rev A02
Draftsman K. GARDNER | Proj Mgr S. YATES

Coventry University

CROWN
Brand-Building Packaging™





MOTORS						
REF	DESCRIPTION	MOTOR SIZE	TYPE	DRIVE SIZE	GROUP	PB STA
M205	WASHER DISCHARGE TABLE CONVEYOR	0.75kW	V/F	1.50 kW	PB201A	A
M206	WASHER DISCHARGE CROSS CONVEYOR BLOWER 1	2.25kW	V/F	7.50 kW	PB203A	C
M207	WASHER DISCHARGE CROSS CONVEYOR BLOWER 2	2.25kW	V/F	7.50 kW		C
M208	MASS RIM COATER INFEED CONVEYOR 1	0.75kW	V/F	1.50 kW	PB205A	D
M209	MASS RIM COATER INFEED CONVEYOR 2	0.75kW	V/F	1.50 kW		D
M210	MASS RIM COATER DISCHARGE CONVEYOR	0.75kW	V/F	1.50 kW		D
M211	BRIGHT CAN VACUUM TURNOVER BELT	2.25kW	V/F	4.00 kW	PB207A	E
M212	BRIGHT CAN VACUUM TURNOVER BLOWER	30.00kW	S/S	45.00 kW	PB208A	F
M215	BRIGHT CAN CONVEYOR 1	0.75kW	V/F	1.50 kW	PB209A	G
M216	BRIGHT CAN CONVEYOR 2	0.75kW	V/F	1.50 kW		G
M218	BRIGHT CAN CONVEYOR 4	0.75kW	V/F	1.50 kW		G
M219	BRIGHT CAN CONVEYOR 5	0.75kW	V/F	1.50 kW		G
M224	BI-DI CROSS CONVEYOR 6	0.75kW	V/F	1.50 kW	PB210A	H
M225	BI-DI LIVE TRANSFER	0.75kW	V/F	1.50 kW		H
M226L	BRIGHT CAN BI-DI LOAD	0.75kW	V/F rev	1.50 kW		H
M226U	BRIGHT CAN BI-DI UNLOAD	0.75kW				H
M227	BRIGHT CAN CONVEYOR 8	0.75kW	V/F	1.50 kW	PB211A	J
M228	BRIGHT CAN CONVEYOR 9	0.75kW	V/F	1.50 kW		J
M229	BRIGHT CAN CONVEYOR 10	0.75kW	V/F	1.50 kW		J
M230	BRIGHT CAN CONVEYOR 11	0.75kW	V/F	1.50 kW		J
M231	BRIGHT CAN CONVEYOR 12	0.75kW	V/F	1.50 kW		J
M235	VACUUM TRANSFER UNIT SUPPLY CONVEYOR (JACK SHAFT)	0.55kW	V/F	1.50 kW	PB212A	K
M236	VACUUM TRANSFER UNIT OVERHEAD BELT	0.55kW	V/F	1.50 kW		K
M237	VACUUM TRANSFER UNIT BLOWER	4.00kW	DOL	4.00 kW	PB214A	L
M238	VACUUM TRANSFER UNIT DISCHARGE CONVEYOR	0.55kW	V/F	1.50 kW		K
M240	BRIGHT CAN AIR SPLITTER BLOWER 1	5.50kW	V/F	5.50 kW		K
M241	DECORATOR 11 AIR CONVEYOR BLOWER 1	4.00kW	DOL	4.00 kW	PB215A	M1
M245	DECORATOR 11 DOUBLING BOX AIR DECK BLOWER	4.00kW	DOL	4.00 kW		M2
M246	DECORATOR 11 DOUBLING BOX BLOWER	7.50kW	DOL	7.50 kW		M3
M247	DECORATOR 11 SINGLE FILER JAM BUSTER	0.37kW	DOL	0.37 kW	PB216A	N
M248	DECORATER 11 AIR INFEED TRACK BLOWER	4.00kW	DOL	4.00 kW		M4
M251	DECORATOR 12 AIR CONVEYOR BLOWER 1	4.00kW	DOL	4.00 kW	P1217A	P1
M252	DECORATOR 12 AIR CONVEYOR BLOWER 2	4.00kW	DOL	4.00 kW		P2
M255	DECORATOR 12 DOUBLING BOX AIR DECK BLOWER	4.00kW	DOL	4.00 kW		P3
M256	DECORATOR 12 DOUBLING BOX BLOWER	7.50kW	DOL	7.50 kW		P4
M257	DECORATOR 12 SINGLE FILER JAM BUSTER	0.37kW	DOL	0.37 kW	PB218A	O
M258	DECORATER 12 AIR INFEED TRACK BLOWER	4.00kW	DOL	4.00 kW		P5

BEACONS & SWITCHES						
REF	DESCRIPTION	TYPE	COLOR	DIST. STA	LOCATION	NOTES
PB204B	MRC EMPTY OUT	ILLUM PB	YELLOW	D201	P201	
PB206X	CAN DUMP SELECTOR SWITCH	SEL SW		D201	P201	TWO-WAY
						SPRING RETURN
PB213Y	CAN DUMP SELECTOR SWITCH	SEL SW		D202	P204	TWO-WAY
						SPRING RETURN
PB219E	DECORATER EMPTY OUT	ILLUM PB	YELLOW	D202	P204	
LT201A	BIDI IN STORAGE	BEACON	RED	D202	B/MEZZ	TOP
LT201B	BIDI IN MANUAL MODE	BEACON	AMBER	D202	B/MEZZ	BASE
LT202B	DECO 12 STANDBY	BEACON	AMBER	D202	B/MEZZ	BASE
LT202B	WASHER DISCH CONV & INVERTER AUTO-RESTART	BEACON	AMBER	D202	B/MEZZ	BASE
LT203A	DECO 11 FAULT	BEACON	RED	D202	B/MEZZ	TOP
LT203B	DECO 11 STANDBY	BEACON	AMBER	D202	B/MEZZ	BASE
LT204A	DECO 12 FAULT	BEACON	RED	D202	B/MEZZ	TOP
LT204B	DECO 12 STANDBY	BEACON	AMBER	D202	B/MEZZ	BASE

SENSORS			
REF	DESCRIPTION	TYPE	DIST. STA
S205A	MRC UV RESTART	PHOTO	D201
S206A	WASHER CRASH STOP	2T	D201
S206B	MRC INFEED (M208) RESTART - HIGH LEVEL	2T	D201
S206C	MRC INFEED (M208) RESTART - HIGH LEVEL	2T	D201
S207A	MRC INFEED (M208) STOP - LOW LEVEL	2T	D201
S207B	MRC INFEED (M208) STOP - HIGH LEVEL	2T	D201
S211A	VTU BELT MOTION DETECTOR	18mm	D201
S215A	M215 BUILDBACK / ELEVATOR CRASH STOP	2T	D201
S215B	M215 BUILDBACK / ELEVATOR CRASH STOP	2T	D201
S216A	M216 BUILDBACK / MRC INFEED (M209) STOP	2T	D201
S216B	M216 BUILDBACK / MRC INFEED (M209) STOP	2T	D201
S218A	M218 MODULATION	2T	D201
S218B	M218 BUILDBACK	2T	D201
S219A	M219 BUILDBACK / BIDI LOAD START	2T	D201
S219B	BIDI LOAD STOP	2T	D201
S224A	M224 BUILDBACK	2T	D201
S224B	BIDI PRESSURE	2T	D201
S226A	BI DI FULL	PHOTO	D201
S226B	CANMAKERS RESTART	PHOTO	D201
S226C	CANMAKER STOP / WASHER EMPTY OUT	PHOTO	D201
S226D	BI DI EMPTY	PHOTO	D201
S227A	M227 BUILDBACK / BIDI UNLOAD	2T	D201
S228A	M228 BUILDBACK	2T	D201
S229A	M229 MODULATION	2T	D202
S229B	M229 BUILDBACK	2T	D202
S230A	M230 BUILDBACK / DECORATOR RESTART	2T	D202
S231A	M231 BUILDBACK	2T	D202
S235A	METERING PACK CONTROL	2T	D202
S236A	VTU BELT MOTION DETECTOR	18mm	D202
S238A	M238 BUILDBACK / VTU CRASH STOP 1	2T	D202
S238B	M238 BUILDBACK / VTU CRASH STOP 2	2T	D202
S240A	AIR DECK HIGH LEVEL	2T	D202
S240B	MODULATION	2T	D202
S241A	AIR DECK LOW LEVEL / DECORATOR LOW LEVEL	2T	D202
S248A	AIR INFEED TRACK VOID	18mm	D202
S252A	AIR DECK LOW LEVEL / DECORATOR LOW LEVEL	2T	D202
S258A	AIR INFEED TRACK VOID	18mm	D202

Rev.	Date	Revisions	By	Appr
A	12/8/14	ISSUED FOR REFERENCE	KaG	
A01	12/19/14	LINE CONTROL REVISIONS	KaG	

CROWN
Brand - Building Packaging
Project Management & Engineering

SUBSYSTEM 2 FUNCTIONAL LAYOUT
WASHER DISCHARGE TO DECO 11 & 12 INFEED

Plant Custines, France

CAD File Name 1247-602A01- Subsystem 2 Functional Layout.dwg

Scale 1:100	Paper size A1	Date 9/30/14
Design Engr K. GARDNER	Checked By	Plant Name-Project Number CUS-1247
Draftsman K. GARDNER	Proj Mgr S. YATES	Sheet 602 / Rev A01

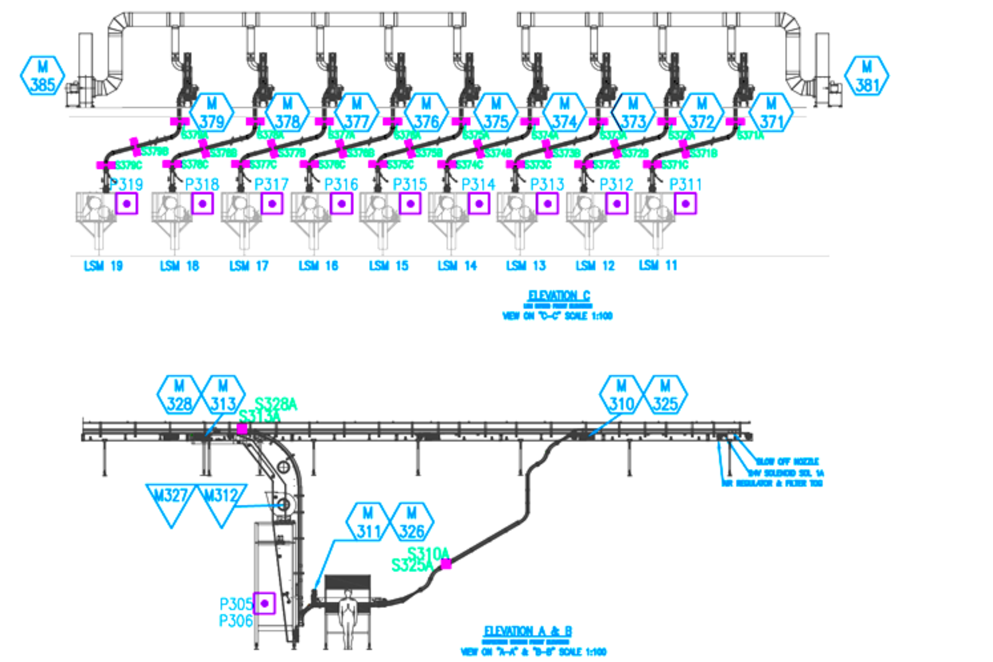
MOTORS						
REF	**DESCRIPTION**	**MOTOR SIZE**	**TYPE**	**DRIVE SIZE**	**GROUP**	**PB STA**
M301	DECO 11 DISCHARGE AIR TRACK BLOWER	5.60kW	DOL		PB301A	A
M302	DECO 11 DISCHARGE AIR TRACK BLOWER	5.60kW	DOL			A
M303	DECO 11 CAN CONVEYOR #1	0.75kW	V/F	1.50kW	PB302A	B
M304	DECO 11 CAN CONVEYOR #2	0.75kW	V/F	1.50kW		B
M305	DECO 11 CAN CONVEYOR #3	0.75kW	V/F	1.50kW		B
M306	DECO 11 CAN CONVEYOR #4	0.75kW	V/F	1.50kW		B
M307	DECO 11 CAN CONVEYOR #5	0.75kW	V/F	1.50kW		B
M308	DECO 11 CAN CONVEYOR #6	0.75kW	V/F	1.50kW	PB303A	C
M310	DECO 11 INSPECTION CONVEYOR	0.75kW	V/F	1.50kW	PB304A	D
M311	DECO 11 LIGHT BOX CONVEYOR	0.375kW	V/F	1.50kW		D
M312	DECO 11 INSPECTION ELEVATOR	1.50kW	V/F	2.20kW		D
M313	DECO 11 INSPECTION ELEVATOR BLOWER	3.75kW	DOL		PB305A	E
M315	DECO 12 DISCHARGE AIR TRACK BLOWER	3.75kW	DOL		PB306A	F
M316	DECO 12 CAN CONVEYOR # 1	0.75kW	V/F	1.50kW	PB307A	G
M317	DECO 12 CAN CONVEYOR # 2	0.75kW	V/F	1.50kW		G
M318	DECO 12 CAN CONVEYOR # 3	0.75kW	V/F	1.50kW		G
M319	DECO 12 CAN CONVEYOR # 4	0.75kW	V/F	1.50kW		G
M320	DECO 12 CAN CONVEYOR # 5	0.75kW	V/F	1.50kW		G
M321	DECO 12 CAN CONVEYOR # 6	0.75kW	V/F	1.50kW	PB308A	H
M322	DECO 12 CAN CONVEYOR # 7	0.75kW	V/F	1.50kW		H
M325	DECO 12 INSPECTION CONVEYOR	0.75kW	V/F	1.50kW	PB309A	J
M327	DECO 12 INSPECTION ELEVATOR	1.50kW	V/F	2.20kW		J
M328	DECO 12 INSPECTION ELEVATOR BLOWER	3.75kW	DOL		PB310A	K
M330	DECORATED CAN CONVEYOR #31	0.75kW	V/F	1.50kW	PB320A	L
M331	DECORATED CAN CONVEYOR #32	0.75kW	V/F	1.50kW		L
M332	BI-DI CROSS CONVEYOR #33	0.75kW	V/F	1.50kW	PB321A	M
M333	BI-DI LIVE TRANSFER	0.75kW	V/F	1.50kW		M
M334L	BI-DI LOAD	0.75kW	V/F rev	1.50kW		M
M334U	BI-DI UNLOAD	0.75kW				M
M336	DECORATED CAN CONVEYOR #34	0.75kW	V/F	1.50kW	PB322A	N
M337	DECORATED CAN CONVEYOR #35	0.75kW	V/F	1.50kW		N
M338	DECORATED CAN CONVEYOR #36	0.75kW	V/F	1.50kW		N
M339	DECORATED CAN CONVEYOR #37	0.75kW	V/F	1.50kW		N
M340	DECORATED CAN CONVEYOR #38	0.75kW	V/F	1.50kW		N
M345	VTU SUPPLY CONVEYOR (JACK SHAFT)	0.75kW	V/F	1.50kW	PB323A	P
M346	VTU OVERHEAD BELT	0.55kW	V/F	0.75kW		P
M347	VTU UNIT BLOWER	4.00kW	DOL	4.00kW	PB324A	Q
M348	VTU DISCHARGE CONVEYOR	0.55kW	V/F	1.50kW		R
M350	LSM MASS CONVEYOR BLOWER #1	4.00kW	V/F	5.50kW	PB325A	S
M351	LSM MASS CONVEYOR BLOWER #2	4.00kW	V/F	5.50kW		S
M352	LSM MASS CONVEYOR BLOWER #3	4.00kW	V/F	5.50kW		S
M353	LSM MASS CONVEYOR BLOWER #4	4.00kW	V/F	5.50kW		S
M361	LSM #11 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW	PB326A	T
M362	LSM #12 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M363	LSM #13 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M364	LSM #14 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M365	LSM #15 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M366	LSM #16 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M367	LSM #17 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M368	LSM #18 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M369	LSM #19 SINGLE FILERS BLOWER	1.50kW	V/F	2.20kW		T
M371	LSM #11 NOSE OVER BELT	0.55kW	V/F	1.50kW	PB327A	U
M372	LSM #12 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M373	LSM #13 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M374	LSM #14 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M375	LSM #15 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M376	LSM #16 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M377	LSM #17 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M378	LSM #18 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M379	LSM #19 NOSE OVER BELT	0.55kW	V/F	1.50kW		U
M381	LSM #11 -14 NOSE OVER BLOWERS	7.50kW	V/F	11.00kW	PB328A	V
M385	LSM #15 -19 NOSE OVER BLOWERS	7.50kW	V/F	11.00kW	PB329A	W
M390	SPRAYED CAN CONVEYOR #1	0.75kW	V/F	1.50kW	PB350A	X
M391	IBO INFEED CROSS CONVEYOR	0.75kW	V/F	1.50kW		X
M392	IBO INFEED LIVE TRANSFER	0.37kW	V/F	1.50kW		X

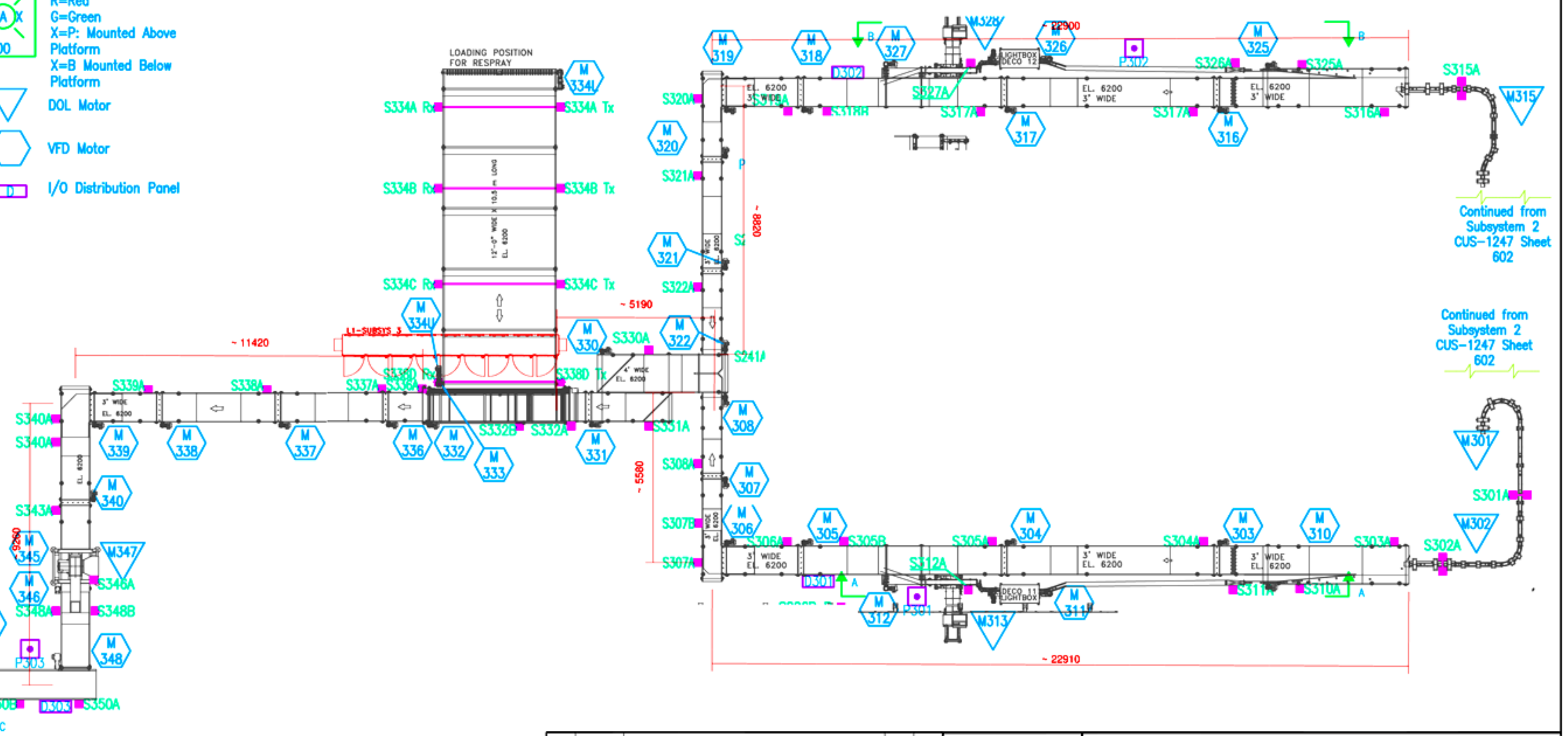
SENSORS			
REF	**DESCRIPTION**	**TYPE**	**DIST. STA**
S301A	CRASH SENSOR	PHOTO	D301
S302A	CRASH SENSOR	PHOTO	D301
S303A	CRASH SENSOR	2T	D301
S304A	M304 BUILDBACK	2T	D301
S305A	M305 BUILDBACK	2T	D301
S305B	DECORATOR STOP / EMPTY OUT	2T	D301
S306A	M306 BUILDBACK	2T	D301
S307A	MODULATION	2T	D301
S307B	M307 BUILDBACK	2T	D301
S308A	M308 BUILDBACK	2T	D301
S310A	JAM SENSOR	2T	D301
S311A	M311 BUILDBACK	2T	D301
S312A	M312 BUILDBACK	2T	D301
S315A	CRASH SENSOR	PHOTO	D302
S316A	CRASH SENSOR	2T	D302
S317A	M317 BUILDBACK	2T	D302
S318A	M318 BUILDBACK	2T	D302
S318B	DECORATOR STOP / EMPTY OUT	2T	D302
S319A	M319 BUILDBACK	2T	D302
S320A	MODULATION	2T	D302
S320B	M320 BUILDBACK	2T	D302
S321A	M321 BUILDBACK	2T	D302
S325A	JAM SENSOR	2T	D302
S326A	M326 BUILDBACK	2T	D302
S327A	M327 BUILDBACK	2T	D302
S330A	M330 BUILDBACK	2T	D303
S331A	M331 BUILDBACK	2T	D303
S331B	BIDI LOAD START	2T	D303
S331C	BIDI LOAD STOP	2T	D303
S332A	M332 BUILDBACK	2T	D303
S332B	BIDI PRESSURE	2T	D303
S334A	BI DI FULL	PHOTO	D303
S334B	BI DI EMPTY	PHOTO	D303
S336A	M336 BUILDBACK / BIDI UNLOAD	2T	D303
S337A	M337 BUILDBACK / BIDI UNLOAD	2T	D303
S338A	M338 BUILDBACK / BIDI UNLOAD	2T	D303
S339A	M339 BUILDBACK / BIDI UNLOAD	2T	D303
S340A	M340 MODULATION	2T	D303
S340B	M340 BUILDBACK	2T	D303
S345A	METERING PACK CONTROL	2T	D303
S346A	VTU BELT MOTION DETECTOR	18mm	D303
S348A	M418 BUILDBACK / VTU CRASH STOP 1	2T	D303
S348B	M418 BUILDBACK / VTU CRASH STOP 2	2T	D303
S350A	AIR DECK HIGH LEVEL / HIGH SPEED	2T	D303
S350B	MODULATION	2T	D303
S350C	AIR DECK LOW LEVEL / LOW SPEED	2T	D303
S371A	NOSE OVER HIGH	PHOTO	D303
S371B	NOSE OVER MODULATION	PHOTO	D303
S371C	LOW LEVEL	PHOTO	D303
S372A	NOSE OVER HIGH	PHOTO	D303
S372B	NOSE OVER MODULATION	PHOTO	D303
S372C	LOW LEVEL	PHOTO	D303
S373A	NOSE OVER HIGH	PHOTO	D303
S373B	NOSE OVER MODULATION	PHOTO	D303
S373C	LOW LEVEL	PHOTO	D303
S374A	NOSE OVER HIGH	PHOTO	D303
S374B	NOSE OVER MODULATION	PHOTO	D303
S374C	LOW LEVEL	PHOTO	D303
S375A	NOSE OVER HIGH	PHOTO	D303
S375B	NOSE OVER MODULATION	PHOTO	D303
S375C	LOW LEVEL	PHOTO	D303
S376A	NOSE OVER HIGH	PHOTO	D303
S376B	NOSE OVER MODULATION	PHOTO	D303
S376C	LOW LEVEL	PHOTO	D303
S377A	NOSE OVER HIGH	PHOTO	D303
S377B	NOSE OVER MODULATION	PHOTO	D303
S377C	LOW LEVEL	PHOTO	D303
S378A	NOSE OVER HIGH	PHOTO	D303
S378B	NOSE OVER MODULATION	PHOTO	D303
S378C	LOW LEVEL	PHOTO	D303
S379A	NOSE OVER HIGH	PHOTO	D303
S379B	NOSE OVER MODULATION	PHOTO	D303
S379C	LOW LEVEL	PHOTO	D303
S390A	IBO CRASH SENSOR	2T	D303

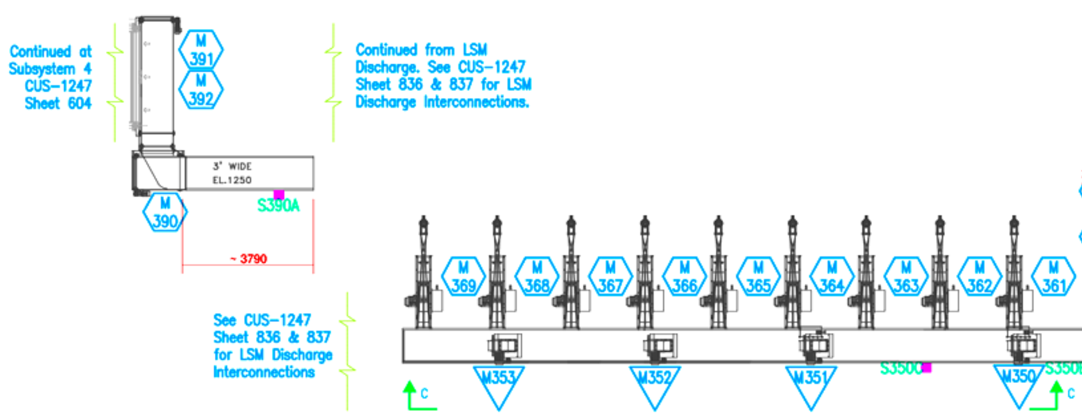
BEACONS & SWITCHES						
REF	**DESCRIPTION**	**TYPE**	**COLOR**	**DIST. STA**	**LOCATION**	**NOTES**
PB315X	SAMPLE CAN BLOW-OFF	ILLUM PB	YELLOW	D301	P301	
SS316X	INCREASE/DECREASE CONVEYOR SPEED	POT		D301	P301	potentiometer
PB317Y	SAMPLE CAN BLOW-OFF	ILLUM PB		D302	P302	
SS318Y	INCREASE/DECREASE CONVEYOR SPEED	POT		D302	P302	potentiometer
SS330X	MANUAL BIDI OPERATION	3 POS SEL		D303	P303	
SS331X	CAN DUMP SELECTOR SWITCH	2 POS SEL		D303	P303	
LT301A	DECORATOR 11 FAULT	BEACON	RED	D301	B / MEZZ	TOP
LT301B	DECORATOR 11 STANDBY	BEACON	AMBER	D301	B / MEZZ	BASE
LT302A	DECO 11 CONVEYOR FAULT	BEACON	RED	D301	B / MEZZ	BASE
LT303A	DECORATOR 12 FAULT	BEACON	RED	D302	B / MEZZ	TOP
LT303B	DECORATOR 12 STANDBY	BEACON	AMBER	D302	B / MEZZ	BASE
LT304A	DECO 12 CONVEYOR FAULT	BEACON	RED	D302	B / MEZZ	BASE
LT305A	LSM INFEED FAULT	BEACON	RED	D303	B/MEZZ	BASE
LT306A	BIDI IN MANUAL MODE	BEACON	BLUE	D303	P303X	BASE
LT307A	LSM AREA FAULT	BEACON	RED	D303	B/MEZZ	BASE
LT308A	IBO HIGH LIMIT	BEACON	RED	D303	B/MEZZ	BASE
LT309A	SPRAYED CAN CONVEYOR AUTO RESTART	BEACON	AMBER	D303	B/MEZZ	BASE

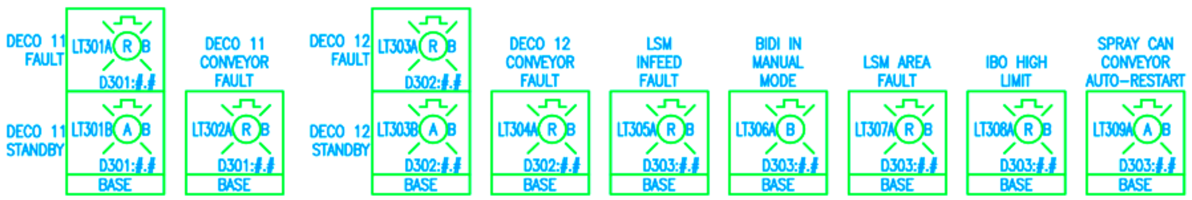
Legend:

- OH IFM
- 1T or 2T or 18mm
- Alarm Horn
- Pressure Switch
- Photo-Eye
- Pushbutton Station
- Blow-off Solenoid (Inspection)
- Beacons: A=Amber, B=Blue, R=Red, G=Green, X=P: Mounted Above Platform, X=B Mounted Below Platform
- DOL Motor
- VFD Motor
- I/O Distribution Panel









Rev.	Date	Revisions	By	Appr
A	12/9/14	ISSUED FOR REFERENCE	KaG	

CROWN Brand-Building Packaging — Project Management & Engineering

LINE 1 SUBSYSTEM 3 FUNCTIONAL LAYOUT
DECO 11 & 12 DISCHARGE TO IBO INFEED

Plant Custines, France
CAD File Name 1247-603A- Subsystem 3 Functional Layout.dwg
Scale 1:100 | Paper size A1 | Date 30 September 2014
Design Engr K. GARDNER | Checked By | Plant Name-Project Number CUS-1247 | Sheet 603 | Rev A
Draftsman K. GARDNER | Proj Mgr S. YATES

Coventry University

CROWN
Brand-Building Packaging™

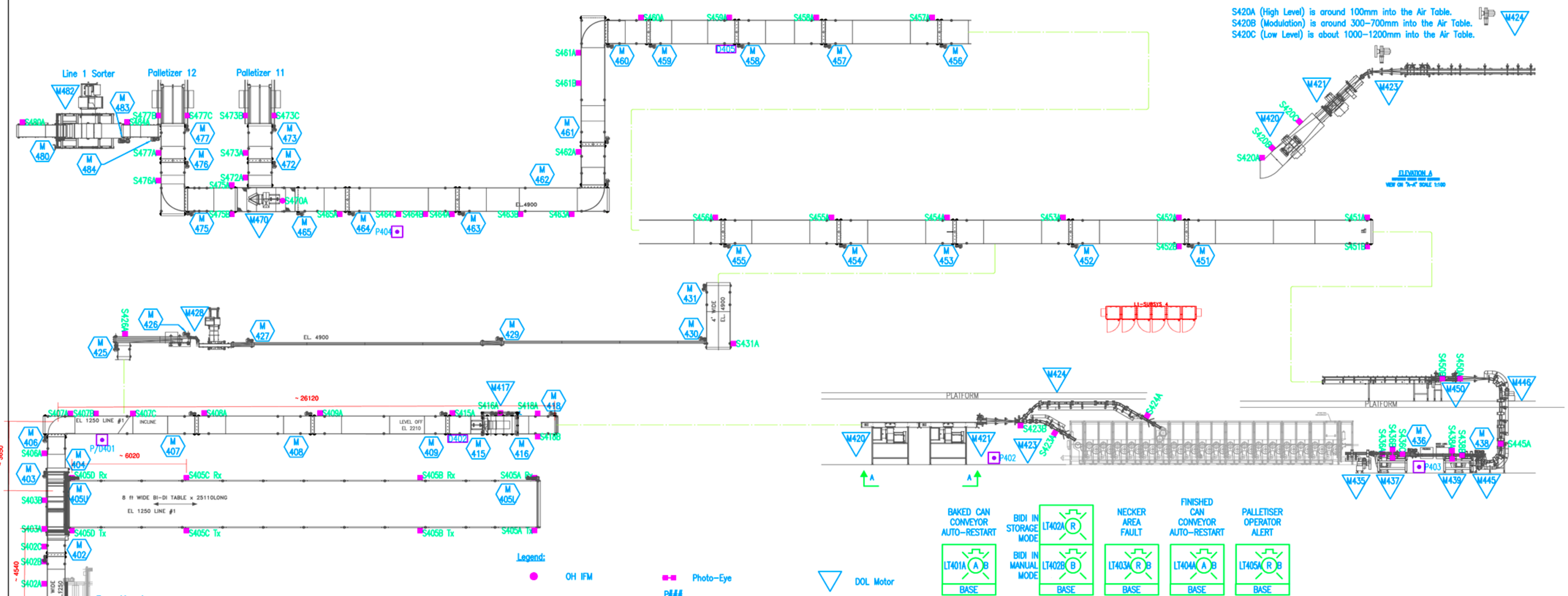
MOTORS						
REF	DESCRIPTION	MOTOR SIZE	TYPE	DRIVE SIZE	GROUP	PB STA
M401	NOT USED					
M402	IBO DISCHARGE CROSS CONVEYOR	0.75kW	V/F	1.50 kW	A	P400 & PD401
M403	BAKED CAN BIDI CROSS CONVEYOR	0.75kW	V/F	1.50 kW	B	P400 & PD401
M404	BI-DI LIVE TRANSFER	0.75kW	V/F	1.50 kW	B	
M405L	BAKED CAN BI-DI LOAD	0.75kW	V/F rev	1.50 kW	B	
M405U	BAKED CAN BI-DI UNLOAD	0.75kW			B	
M406	BAKED CAN CONVEYOR #1	0.75kW	V/F	1.50 kW	C	P400,PD401,P402
M407	BAKED CAN CONVEYOR #2	0.75kW	V/F	1.50 kW	C	
M408	BAKED CAN CONVEYOR #3	0.75kW	V/F	1.50 kW	C	
M409	BAKED CAN CONVEYOR #4	0.75kW	V/F	1.50 kW	C	
M415	VACUUM TRANSFER UNIT SUPPLY CONVEYOR (JACK SHAFT)	0.75kW	V/F	1.50 kW	D	P400 & P402
M416	VACUUM TRANSFER UNIT OVERHEAD BELT	0.75kW	V/F	1.50 kW	D	
M417	VACUUM TRANSFER UNIT BLOWER	3.75kW	DOL		E	P400 & P402
M418	VACUUM TRANSFER UNIT DISCHARGE CONVEYOR	0.75kW	V/F	1.50 kW	D	P400 & P402
M420	NECKER CONVERGING RAIL ASF BLOWER #1	3.75kW	DOL		F	P400 & P402
M421	NECKER CONVERGING RAIL ASF BLOWER #2	3.75kW	DOL		F	
M422	NOT USED					
M423	NECKER INFEED TRACK BLOWER #1	5.60kW	DOL		F	
M424	NECKER INFEED TRACK BLOWER #2	5.60kW	DOL		F	
M425	RESPRAY/ NECKER BYPASS CONVEYOR	0.75kW	V/F	1.50 kW	G	P400 & PD401
M426	VACUUM CONVEYOR ELEVATOR BELT	0.75kW	V/F	1.50 kW	G	
M427	RESPRAY SINGLE LANE CONVEYOR #1	0.75kW	V/F	1.50 kW	G	
M428	VACUUM CONVEYOR BLOWER	3.75kW	DOL		H	P400 & PD401
M429	RESPRAY SINGLE LANE CONVEYOR #2	0.75kW	V/F	1.50 kW	G	P400 & PD401
M430	RESPRAY SINGLE LANE CONVEYOR #3	0.75kW	V/F	1.50 kW	G	
M431	RESPRAY /NECKER BYPASS REFEED CONVEYOR	0.75kW	V/F	1.50 kW	G	
M435	NECKER DISCHARGE AIR BOOT	3.75kW	DOL		J	P400 & P403
M436	HORIZONTAL INSPECTION CONVEYOR #1	1.50kW	V/F	2.20kW	K	P400 & P403
M437	HORIZONTAL INSPECTION BLOWER #1	11.25kW	DOL		J	P400 & P403
M438	HORIZONTAL INSPECTION CONVEYOR #2	1.50kW	V/F	2.20kW	K	P400 & P403
M439	HORIZONTAL INSPECTION BLOWER #2	11.25kW	DOL		J	P400 & P403
M445	AIR ELEVATOR BLOWER #1	3.75kW	DOL		J	
M446	AIR ELEVATOR BLOWER #2	3.75kW	DOL		J	

MOTORS						
REF	DESCRIPTION	MOTOR SIZE	TYPE	DRIVE SIZE	GROUP	PB STA
M450	FINISHED CAN CONVEYOR #1 (SLOW DOWN)	3.75kW	DOL		L	P400,P403,P404
M451	FINISHED CAN CONVEYOR #2	0.75kW	V/F	1.50 kW	L	
M452	FINISHED CAN CONVEYOR #3	0.75kW	V/F	1.50 kW	L	
M453	FINISHED CAN CONVEYOR #4	0.75kW	V/F	1.50 kW	L	P400 & P404
M454	FINISHED CAN CONVEYOR #5	0.75kW	V/F	1.50 kW	M	
M455	FINISHED CAN CONVEYOR #6	0.75kW	V/F	1.50 kW	M	
M456	FINISHED CAN CONVEYOR #7	0.75kW	V/F	1.50 kW	M	
M457	FINISHED CAN CONVEYOR #8	0.75kW	V/F	1.50 kW	M	
M458	FINISHED CAN CONVEYOR #9	0.75kW	V/F	1.50 kW	M	
M459	FINISHED CAN CONVEYOR #10	0.75kW	V/F	1.50 kW	M	
M460	FINISHED CAN CONVEYOR #11	0.75kW	V/F	1.50 kW	M	
M461	FINISHED CAN CONVEYOR #12	0.75kW	V/F	1.50 kW	M	
M462	FINISHED CAN CONVEYOR #13	0.75kW	V/F	1.50 kW	M	
M463	FINISHED CAN CONVEYOR #14	0.75kW	V/F	1.50 kW	M	
M464	FINISHED CAN CONVEYOR #15	0.75kW	V/F	1.50 kW	M	
M465	FINISHED CAN CONVEYOR #16	0.75kW	V/F	1.50 kW	M	
M466	FUTURE	0.75kW	V/F	1.50 kW	M	
M467	FUTURE	0.75kW	V/F	1.50 kW	M	
M470	PALLETISER AIR SPLITTER BLOWER	3.75kW	V/F	7.50kW	P	P400 & P404
M471	FUTURE	0.75kW	V/F	1.50 kW	Q	P400 & P404
M472	PALLETISER 11 FEED CONVEYOR #1	0.75kW	V/F	1.50 kW	Q	
M473	PALLETISER 11 FEED CONVEYOR #2	0.75kW	V/F	1.50 kW	Q	
M475	PALLETISER 12 METERING CONVEYOR	0.75kW	V/F	1.50 kW	R	P400 & P404
M476	PALLETISER 12 FEED CONVEYOR #1	0.75kW	V/F	1.50 kW	R	
M477	PALLETISER 12 FEED CONVEYOR #2	0.75kW	V/F	1.50 kW	R	
M480	SORTED CAN CONVEYOR #1	0.75kW	V/F	1.50 kW	S	P400 & P404
M481	NOT USED					
M482	SORTED CAN ELEVATOR	1.50kW	V/F	2.20 kW	S	P400 & P404
M483	SORTED CAN ELEVATOR BLOWER	11.25kW	DOL		T	P400 & P404
M484	SORTED CAN ELEVATOR DISCHARGE CONVEYOR	0.75kW	V/F	1.10kW	S	P400 & P404

SENSORS							
REF	DESCRIPTION	TYPE	DIST. STA	REF	DESCRIPTION	TYPE	DIST. STA
S402A	M402 BUILDBACK / IBO CRASH STOP	2T	PD401	S452A	M452 BUILDBACK	2T	D404
S402B	BIDI LOAD START	2T	PD401	S452B	M452 BUILDBACK	2T	D404
S402C	BIDI LOAD STOP	2T	PD401	S453A	M453 BUILDBACK	2T	D404
S403A	M403 BUILDBACK	2T	PD401	S454A	M454 BUILDBACK	2T	D404
S403B	BIDI PRESSURE	2T	PD401	S455A	M455 BUILDBACK	2T	D404
S405A	BI DI FULL	PHOTO	PD401	S456A	M456 BUILDBACK	2T	D404
S405B	LSM FEED STOP RESET	PHOTO	PD401	S457A	M457 BUILDBACK	2T	D404
S405C	LSM FEED STOP / IBO EMPTY OUT	PHOTO	PD401	S458A	M458 BUILDBACK	2T	D404
S405D	BI DI EMPTY	PHOTO	PD401	S459A	M459 BUILDBACK	2T	D405
S406A	M406 BUILDBACK / BIDI UNLOAD	2T	PD401	S460A	M460 BUILDBACK	2T	D405
S407A	M407 MODULATION	2T	PD401	S461A	M461 BUILDBACK	2T	D405
S407B	M407 BUILDBACK	2T	PD401	S461B	M461 MODULATION	2T	D405
S407C	NECKER BYPASS GATE	18mm	PD401	S462A	M462 BUILDBACK	2T	D405
S408A	M408 BUILDBACK	2T	PD401	S463A	M463 BUILDBACK	2T	D405
S409A	M409 BUILDBACK	2T	PD401	S463B	M463 MODULATION	2T	D405
S415A	METERING PACK CONTROL	2T	D402	S464A	M464 BUILDBACK	2T	D405
S416A	VTU BELT MOTION DETECTOR	18mm	D402	S464B	PALLETISER FEED RESTART	2T	D405
S418A	M418 BUILDBACK / VTU CRASH STOP 1	2T	D402	S464C	PALLETISER FEED STOP	2T	D405
S418B	M418 BUILDBACK / VTU CRASH STOP 2	2T	D402	S465A	M465 BUILDBACK	2T	D405
S420A	AIR DECK HIGH LEVEL / HIGH SPEED	2T	D402	S466A	M466 BUILDBACK	2T	D405
S420B	MODULATION	2T	D402	S467A	M467 BUILDBACK	2T	D405
S420C	AIR DECK LOW LEVEL / LOW SPEED	2T	D402	S470A	M470 BUILDBACK	OH	D405
S423A	AIR INFEED TRACK VOID	18mm	D402	S471A	M471 BUILDBACK	2T	D405
S423B	INFEED TRACK GATE	2T	D402	S471B	M471 BUILDBACK	2T	D405
S424A	ALTERNATE INFEED GATE	18mm	D402	S472A	M472 BUILDBACK	2T	D405
S426A	M426 JAM SENSOR	2T	D402	S473A	M473 BUILDBACK	2T	D405
S431A	M431 BUILDBACK	2T	PD401	S473A	PALLETISER 11 HIGH LIMIT #1	2T	D405
S436A	CRASH SENSOR	1T	D404	S473A	PALLETISER 11 HIGH LIMIT #2	2T	D405
S436B	COUNT IN SENSOR	PHOTO	D404	S475A	M475 BUILDBACK	2T	D405
S436C	MOTION DETECTION SENSOR	18mm	D404	S475B	M475 BUILDBACK	2T	D405
S438A	COUNT OUT SENSOR	PHOTO	D404	S476A	M476 BUILDBACK	2T	D405
S438B	MOTION DETECTION SENSOR	18mm	D404	S477A	M477 BUILDBACK	2T	D405
S445A	CRASH SENSOR	1T	D404	S477B	PALLETISER 12 HIGH LIMIT #1	2T	D405
S450A	M450 BUILDBACK / CRASH STOP	2T	D404	S477C	PALLETISER 12 HIGH LIMIT #2	2T	D405
S450B	M450 BUILDBACK / CRASH STOP	2T	D404	S480A	M480 BUILDBACK	2T	D405
S451A	M451 BUILDBACK	2T	D404	S484A	M484 BUILDBACK	2T	D405
S451B	M451 BUILDBACK	2T	D404				

BEACONS & SWITCHES						
REF	DESCRIPTION	TYPE	COLOR	DIST. STA	LOCATION	NOTES
SS403X	MANUAL BIDI OPERATION	SEL SW		D402	P402	3-POSITION
						SPRING RET
						CENTER
SS406X	CAN DUMP SELECTOR SWITCH	SEL SW		D402	P402	2-POSITION
						SPRING RET
PB409X	NECKER EMPTY OUT	ILLUM PB	YELLOW	D402	P402	
SS412X	INFEED TRACK SELECTOR SW	SEL SW		D402	P402	2-POSITION
						MAINTAINED
SS413X	NECKER BYPASS SELECTOR SW	SEL SW		D402	P402	2-POSITION
						MAINTAINED
PB437A	PALLETISER EMPTY OUT	ILLUM PB	YELLOW	D405	P405	
LT401A	BAKED CAN CONVEYOR AUTO RESTART	BEACON	AMBER	PD401	PD401	BASE
LT402A	BIDI IN STORAGE MODE	BEACON	RED	PD401	PD401	TOP
LT402B	BIDI IN MANUAL MODE	BEACON	BLUE	PD401	PD401	BASE
LT403A	NECKER AREA FAULT	BEACON	RED	P402	D402	BASE
LT404A	FINISHED CAN CONVEYOR AUTO-RESTART	BEACON	AMBER	PD401	PD401	BASE
LT405A	PALLETISER OPERATOR ALERT	BEACON	RED	PD401	PD401	BASE

S420A (High Level) is around 100mm into the Air Table.
S420B (Modulation) is around 300–700mm into the Air Table.
S420C (Low Level) is about 1000–1200mm into the Air Table.



Legend:

- OH IFM
- 1T or 2T or 18mm
- Alarm Horn
- Pressure Switch
- Subsystem Panel
- Photo-Eye
- Pushbutton Station
- Blow-off Solenoid (Inspection)
- Beacons: X=A→Amber, X=B→Blue, X=R→Red, X=G→Green, Y=P→Mounted on Panel, Y=B→Mounted Below Platform (BASE)
- DOL Motor
- VFD Motor
- Soft Starter Motor
- I/O Distribution Panel
- Pushbutton/ Distribution Station

Rev.	Date	Revisions	By	Appr
A	11/14/14	ISSUED FOR REFERENCE.	KaG	IB
A01	11/20/14	ADDED MOTOR NUMBERING FOR SORTER AREA.	KaG	WS
A02	12/22/14	REVISED PALLETISER AREA, RENUMBERING.	KaG	WS

CROWN Brand-Building Packaging
Project Management & Engineering

LINE 1 SUBSYSTEM 4 FUNCTIONAL LAYOUT IBO DISCHARGE TO PALLETIZER

Plant Custines, France

CAD File Name 1247-604A02- Subsystem 4 Functional Layout.dwg

Scale 1:100 | Paper size A1 | Date 9/30/14

Design Engr K. GARDNER | Checked By | Plant Name-Project Number CUS-1247 | Sheet 604 | Rev A01

Draftsman K. GARDNER | Proj Mgr S. YATES

Appendix 4
Large Server

Use

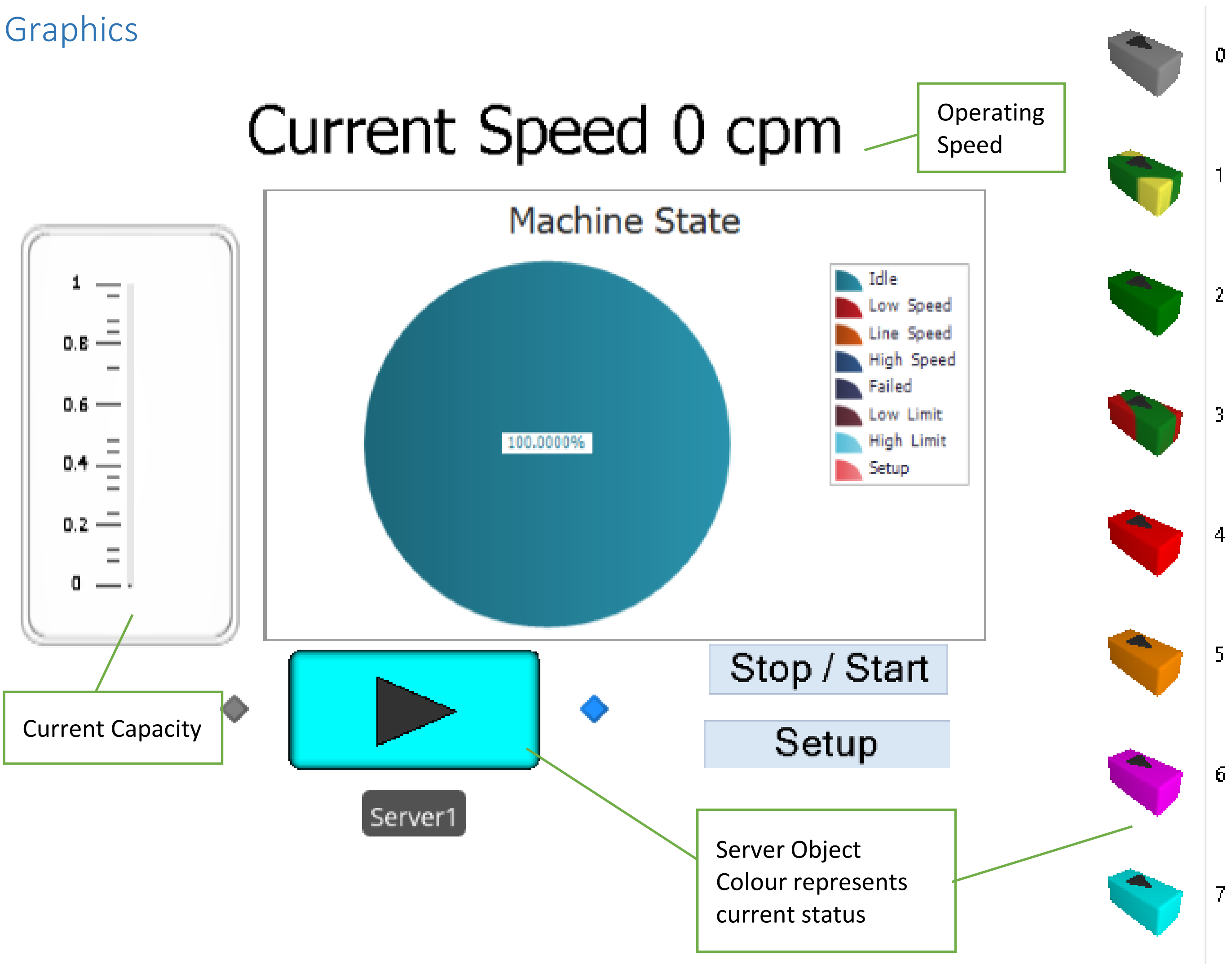
The large server object is used in place of a standard server. It can be “right-click” swapped in place of an existing server in a model. The server is designed to represent servers with capacity >100. The object scales capacity as opposed to process time and therefore scale is not present in the process time calculation.

$$Process\ Time\ (\text{min}) = \frac{Capacity}{Speed(cpm)}$$

The capacity of the server is scaled in accordance with model (or even conveyor) scales and the and therefore would be a problem for servers where the capacity is not much larger than the scale of the model.

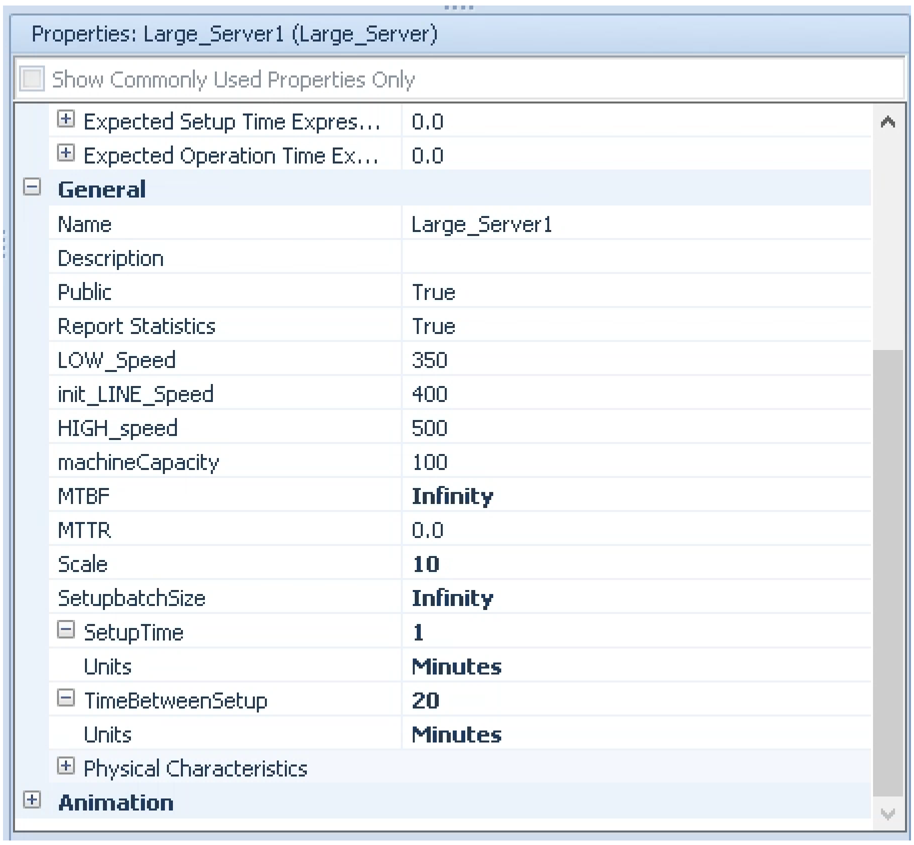
The larger server supports setup options for batch size or time to next changeover. Both the batch size and time between setup support equation entry to allow for variation. The setup time also supports equation entry for variation.

Graphics



- The graphics include status to show how the:
- Operating Speed – It is possible to override the objects line speed and therefore the current running speed is always shown.
- Status history – via a pie chart to show how much time the machine has spent in a state
- The Server graphic – the standard Simio cuboid with the current state of the machine being represented as a colour.
   0. Idle – grey
   1. Low speed – Green with yellow corners
   2. Line Speed – Green
   3. High Speed – Green with red corners
   4. Failed - Red
   5. Low Limit – Orange
   6. High Limit – Purple
   7. Setup – light blue

Data Entry



The extra data required by the large server is defined in the sub model as properties which in the overall model show for entry in the "General" section of the properties window. The extra data required for the server is:

- LowSpeed – The minimum operating speed of the machine in cans per minute (cpm)
- InitLineSpeed – the standard operating speed in cpm. Where the line speed is modified, it serves as the maximum line speed of the machine
- HighSpeed – the maximum operating speed in cpm.
- Machine capacity – This allows for multi-head processes. This number scales up the processing time and therefore if the numbers exceed 50, the large server should be used.
- The server supports processing time based failure which is entered as an expressions for uptime between failure and time to repair. Default MTBF=Infinity means there are no failures

- Scale – This is used to scale the server and should be consistent with overall model scales.
- SetupBatchSize – an expression value for the size of a batch before setup is required. Set to infinity means that no batching is required
- SetupTime – An expression value of the time to change over the machine
- TimeBetweenSetup – as an alternative to counting for batches, the machine can go into setup state based on time. Note this is simulation time, not server running time.

Data Output

The server has all the standard server statistics with the addition of:

- DailyCount – User defined tally statistic element. The daily output of the server is counted. This value is scaled back up so that no further calculation is required.
- Speed – User defined status variable to record the time that the server spends in a given state. The standard server status variable does not adequately represent the extra states required for Crown production lines. A custom status variable "speed" includes the extra speeds that can be set via events from the overall model.

Value	Inbuilt Status Variable	Custom Status variable	Value
0	Starved	Idle	0
		Low limit	5
1	Processing	Low Speed	1
		Line Speed	2
		High Speed	3
2	Blocked	High Limit	6
3	Failed	Failed	4
4	OffShift	n/a	
5	Failed Processing	n/a	
6	Setup	n/a	
7	Off Shift Processing	n/a	
8	Off Shift Setup	n/a	

Timers

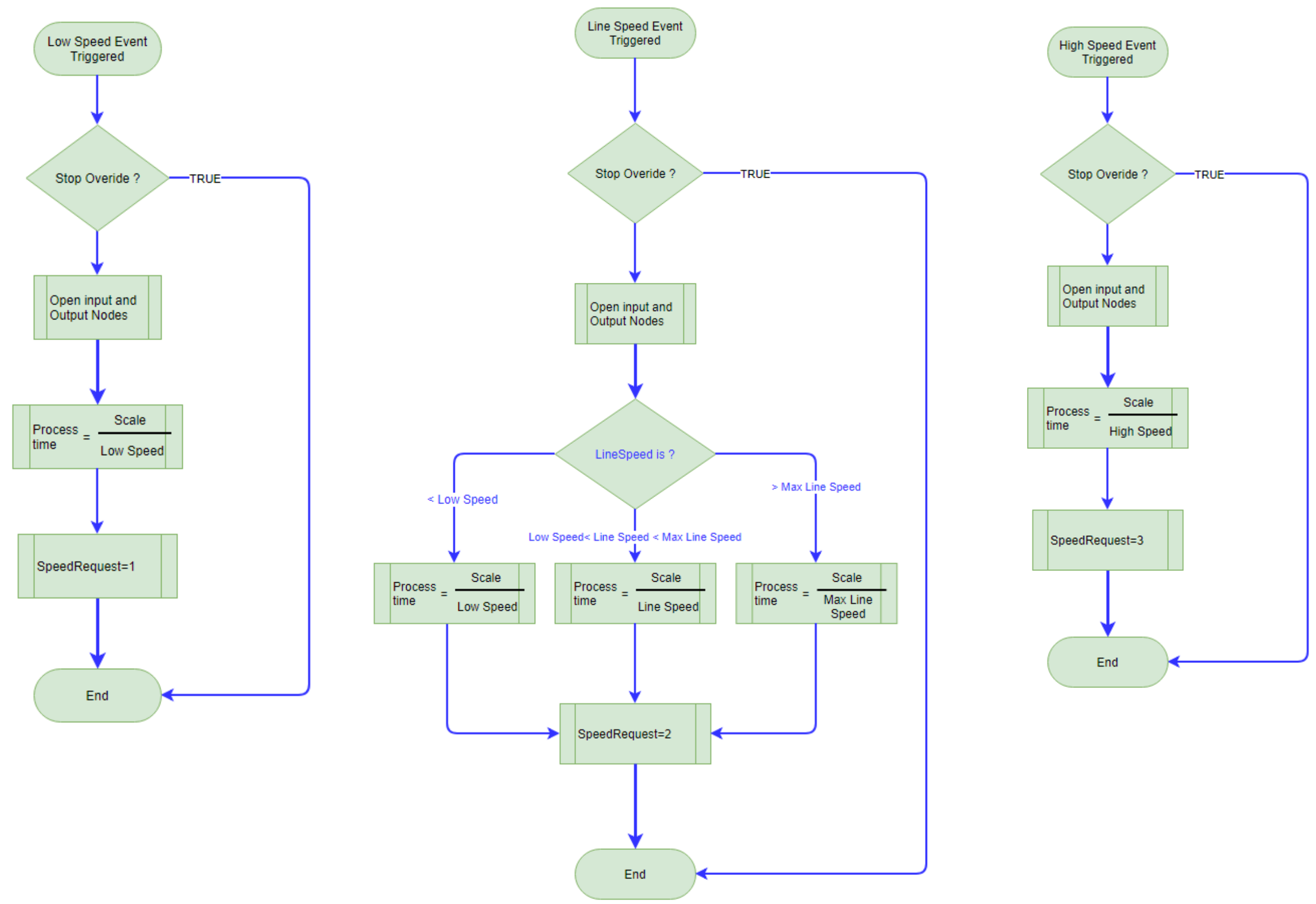
Timer	Function	Triggers Event
Day_Timer	Fires event to record throughput for 24hr period	DayTimerEvent
BatchTimer	Fires event based on time as set by "TimeBetweenSetup" expression	SetupProcess

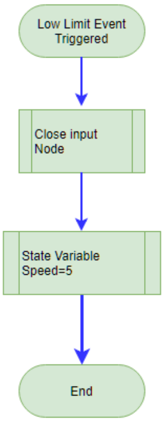
Events

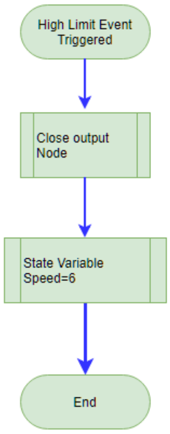
Custom events are used to control the server which are visible (i.e. are able to be triggered) by the overall model. Some private events are used to update the status variable as per the current speed request

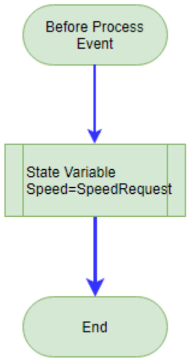
Event	Action	Public
setSpeed_HIGH	Sets the server to run at the speed as input “HighSpeed”	True
setSpeed_LINE	Sets the server to run at the speed as input “LineSpeed”	True
setSpeed_LOW	Sets the server to run at the speed as input “LowSpeed”	True
SetSpeed_stop_HiLim	Stops the server with status high limit	True
SetSpeed_stop_LoLim	Stops the server with status low limit	True
Before Processing	Sets state variable speed per speed request	False
After Processing	Sets state variable speed per speed request	False
On Failed	Sets Alive=0 Sets state variable speed=4	False
On Repaired	Sets Alive=1 Sets status variable speed=speedRequest	False
StopStart	Stops or starts server – used from button on GUI Stops server with speed status=0 Starts server with status variable speed=speedRequest	False
Day_TimerEvent	Calculates the throughput for 24 hr period	False
SetupProcess	Stops server with status variable speed=7 Delays for time setupTime Starts server with status variable speed=speedRequest	Fasle

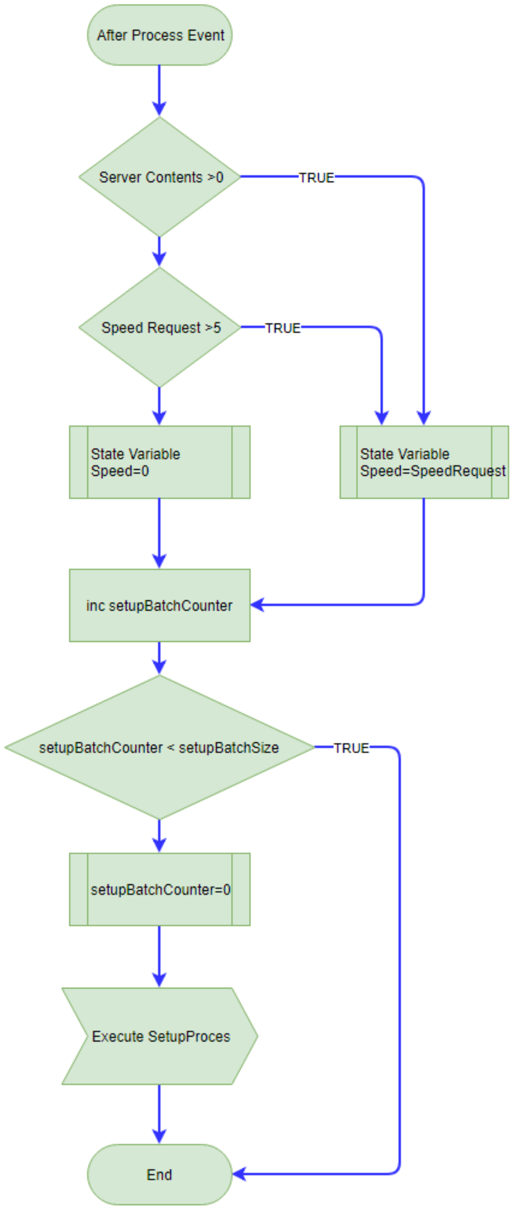
Event Logic

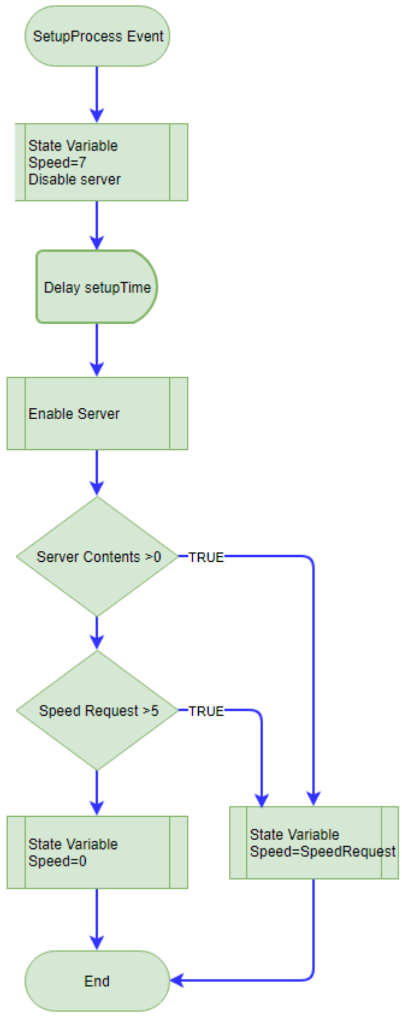


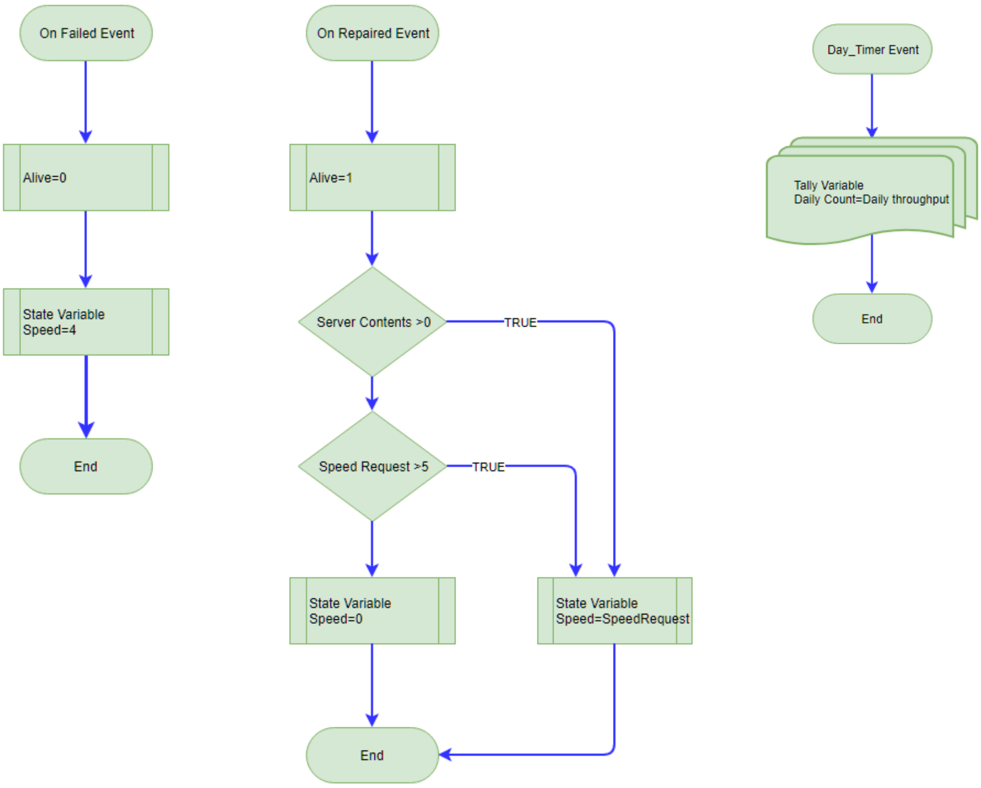

Low Limit Event Triggered
Close input Node
State Variable Speed=5
End


High Limit Event Triggered
Close output Node
State Variable Speed=6
End


Before Process Event
State Variable Speed=SpeedRequest
End


After Process Event
Server Contents >0
TRUE
Speed Request >5
TRUE
State Variable Speed=0
State Variable Speed=SpeedRequest
inc setupBatchCounter
setupBatchCounter < setupBatchSize
TRUE
setupBatchCounter=0
Execute SetupProces
End


SetupProcess Event
State Variable Speed=7 Disable server
Delay setupTime
Enable Server
Server Contents >0
TRUE
Speed Request >5
TRUE
State Variable Speed=0
State Variable Speed=SpeedRequest
End


On Failed Event
Alive=0
State Variable
Speed=4
End
On Repaired Event
Alive=1
Server Contents >0
TRUE
Speed Request >5
TRUE
State Variable
Speed=0
State Variable
Speed=SpeedRequest
End
Day_Timer Event
Tally Variable
Daily Count=Daily throughput
End

Appendix 5
Small Server

Use

The small server is used in place of a standard server. It can be “right-click” swapped in place of an existing server in a model.

The server calculates the processing time in accordance with the scale and capacity.

$$Process\ Time\ (\text{min}) = \frac{Scale * Capacity}{Speed(cpm)}$$

If the capacity is greater than 1, then each cycle, “Capacity” cans will be processed and therefore the capacity is effectively cancelled out. This is not a problem for the server such as body makers, trimmers with single capacity but becomes a problem, if used to represent an industrial oven with a capacity ~30,000. The processing time becomes too large, in the order of hours, rather than minutes and chokes the whole simulation.

Graphics

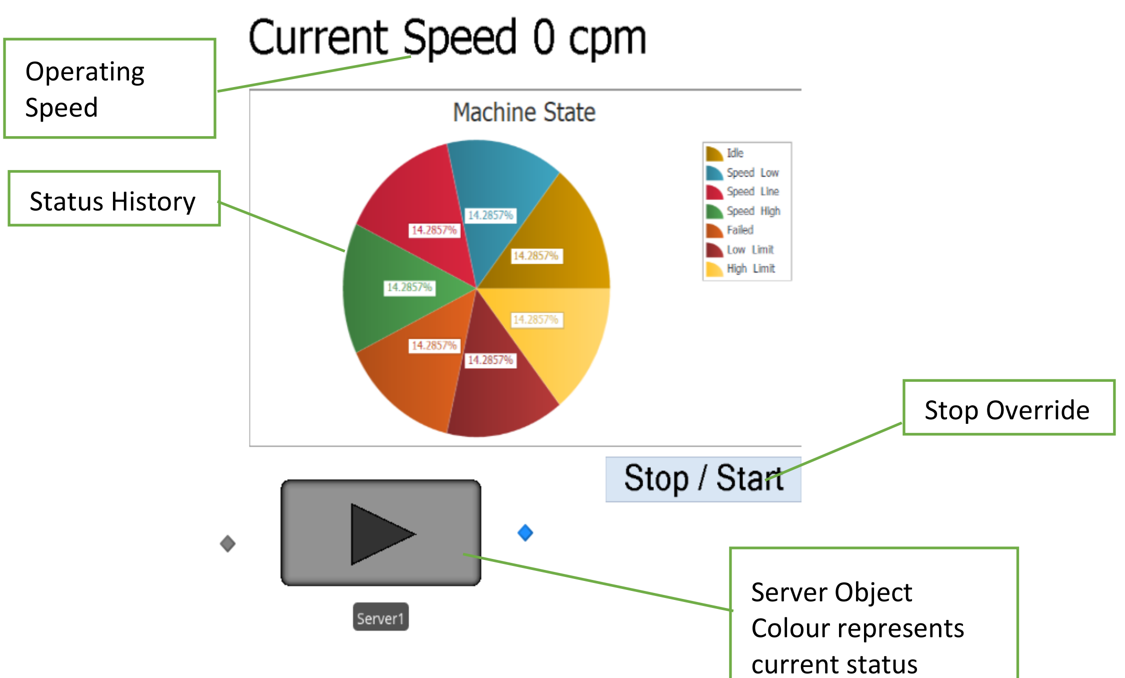
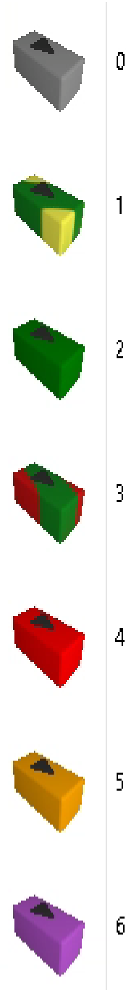


Figure 1Small Server Representation



The graphics include status to show how the:

Operating Speed – It is possible to override the objects line speed and therefore the current running speed is always shown.

Status history – via a pie chart to show how much time the machine has spent in a state

The Server graphic – the standard Simio cuboid with the current state of the machine being represented as a colour.

0. Idle – grey
1. Low speed – Green with yellow corners
2. Line Speed – Green
3. High Speed – Green with red corners
4. Failed - Red
5. Low Limit – Orange
6. High Limit – Purple

Data Entry

The extra data required by the small server is defined in the sub model as properties which show for entry in the "General" section of the properties window. The extra data required for the server is:

- LowSpeed – The minimum operating speed of the machine in cans per minute (cpm)
- InitLineSpeed – the standard operating speed in cpm. Where the line speed is modified, it serves as the maximum line speed of the machine
- HighSpeed – the maximum operating speed in cpm.
- Machine capacity – This allows for multi-head processes. This number scales up the processing time and therefore if the numbers exceed 50, the large server should be used.
- The server supports processing time based failure which is entered as an expressions for uptime between failure and time to repair
- Scale – This is used to scale the server process time and should be consistent with overall model scales.

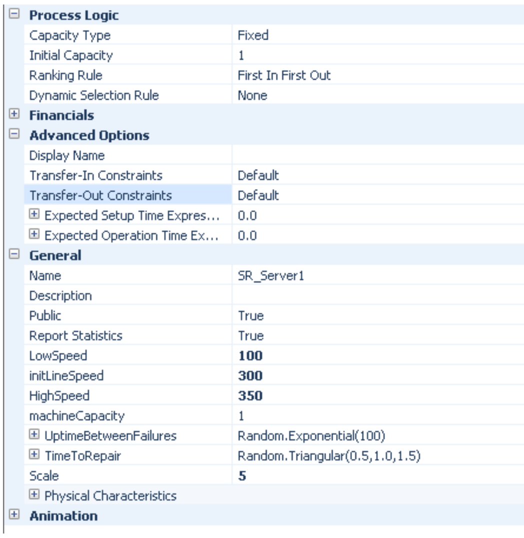


Figure 2 Data Entry

Data Output

The server has all the standard server statistics with the addition of:

- DailyCount – User defined tally statistic element. The daily output of the server is counted. This value is scaled back up so that no further calculation is required.
- Speed – User defined status variable to record the time that the server spends in a given state. The standard server status variable does not adequately represent the extra states required for Crown production lines. A custom status variable "speed" includes the extra speeds that can be set via events from the overall model.

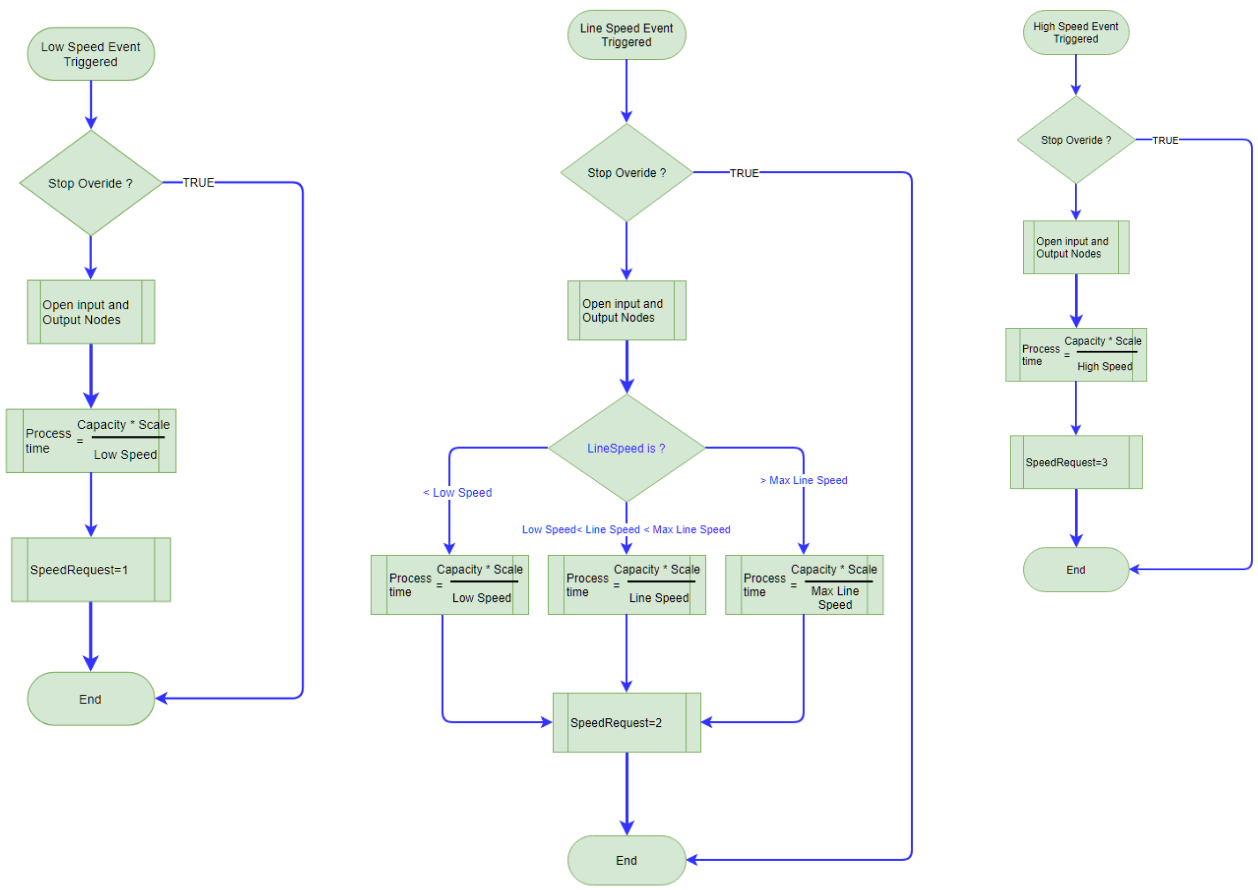
Value	Inbuilt Status Variable	Custom Status variable	Value
0	Starved	Idle	0
		Low limit	5
1	Processing	Low Speed	1
		Line Speed	2
		High Speed	3
2	Blocked	High Limit	6
3	Failed	Failed	4
4	OffShift	n/a	
5	Failed Processing	n/a	
6	Setup	n/a	
7	Off Shift Processing	n/a	
8	Off Shift Setup	n/a	

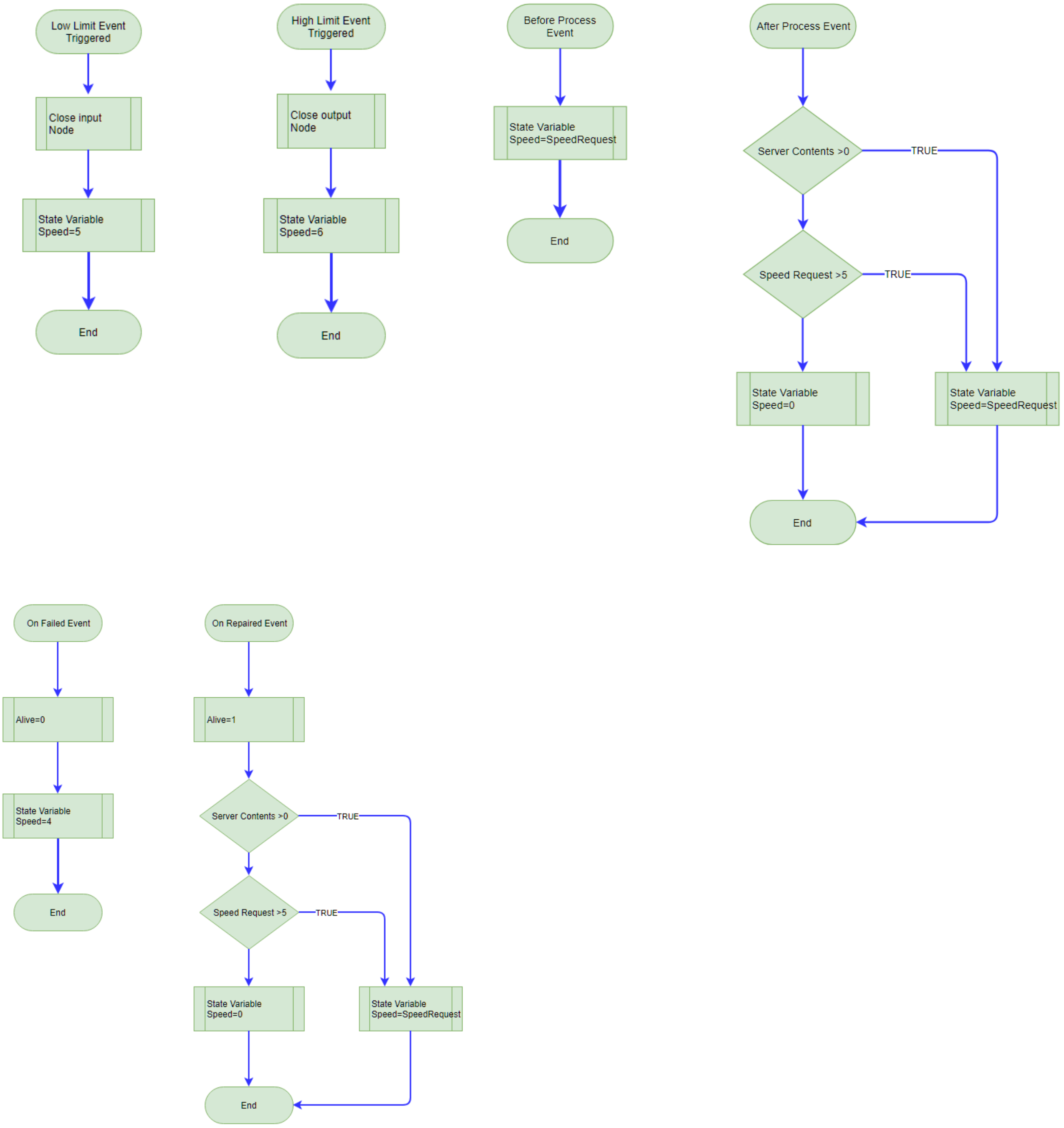
Events

Custom events are used to control the server which are visible (i.e. are able to be triggered) by the overall model. Some private events are used to update the status variable as per the current speed request

Event	Action	Public
setSpeed_HIGH	Sets the server to run at the speed as input “HighSpeed”	True
setSpeed_LINE	Sets the server to run at the speed as input “LineSpeed”	True
setSpeed_LOW	Sets the server to run at the speed as input “LowSpeed”	True
SetSpeed_stop_HiLim	Stops the server with status high limit	True
SetSpeed_stop_LoLim	Stops the server with status low limit	True
Before Processing	Sets state variable speed per speed request	False
After Processing	Sets state variable speed per speed request	False
On Failed	Sets Alive=0 Sets state variable speed=4	False
On Repaired	Sets Alive=1 Sets status variable speed=speedRequest	False
StopStart	Stops or starts server – used from button on GUI Stops server with speed status=0 Starts server with status variable speed=speedRequest	False

Event Logic




Low Limit Event Triggered
Close input Node
State Variable Speed=5
End
High Limit Event Triggered
Close output Node
State Variable Speed=6
End
Before Process Event
State Variable Speed=SpeedRequest
End
After Process Event
Server Contents >0
TRUE
Speed Request >5
TRUE
State Variable Speed=0
State Variable Speed=SpeedRequest
End
On Failed Event
Alive=0
State Variable Speed=4
End
On Repaired Event
Alive=1
Server Contents >0
TRUE
Speed Request >5
TRUE
State Variable Speed=0
State Variable Speed=SpeedRequest
End

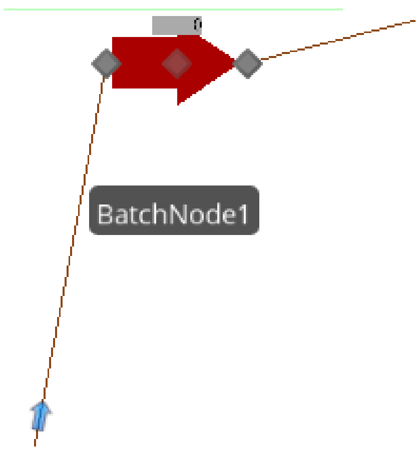
Appendix 6
Batch / Un-Batch Node

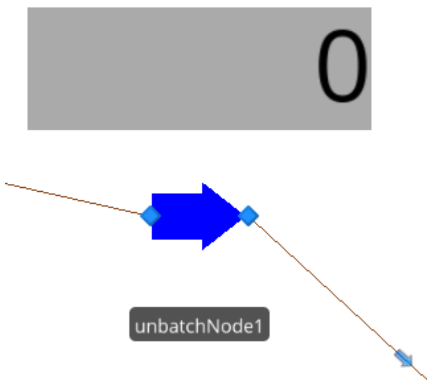
Use

The batch and unbatch nodes are used to further reduce the active entities in the system by batching entities on to conveyors. The use of batching allows for a smaller scale and therefore more accuracy through the smaller servers. The reduction of active entities mean that there are fewer events in the event queue and therefore a reduction in the simulation execution time.

Where a batchnode is placed in the model, the subsequent stations and conveyors should be scaled by the product of the overall model scale and the BatchNode's batchScalar value.

Graphics



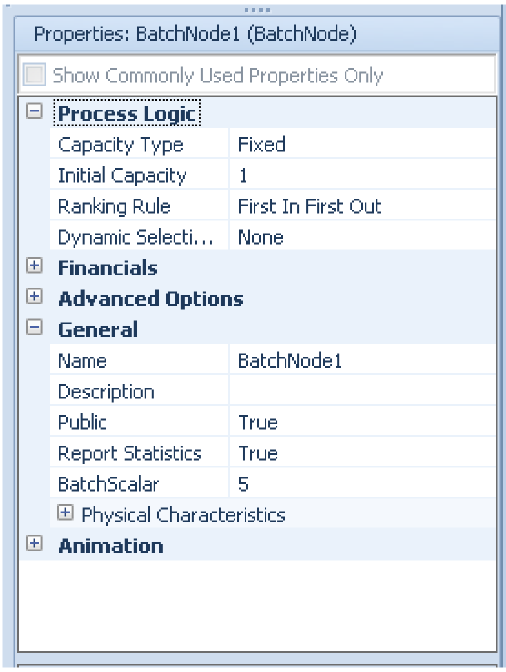


Data Entry

The data in the General section can be replace by variables or data table values and therefore the conveyor properties can be driven from Excel. The extra data required for the batchNode is:

- BatchScalar – The number of entities in a batch for the next conveyor or station.

There is no data required for the unBatchNode.



Data Output

There is no extra output above the output generated by the BatchNode and unBatchNodes.

Appendix 7
CrownConveyor

Use

The conveyor is very useful in showing queues and gaps on the conveying network. The standard conveyor cannot have a variable in place of the logical length and therefore cannot be drives from an Excel table.

The CrownConveyor is sub class of the Simio conveyor. It has inputs:

1. logical length – The real system length of the conveyor.
2. Logical speed – The real system conveyor speed.
3. Capacity – the per metre capacity of the real system.
4. Scale – The number of cans represented by one entity on the conveyor

On initialisation, it scales the conveyor speed and sets the maximum capacity of the conveyor:

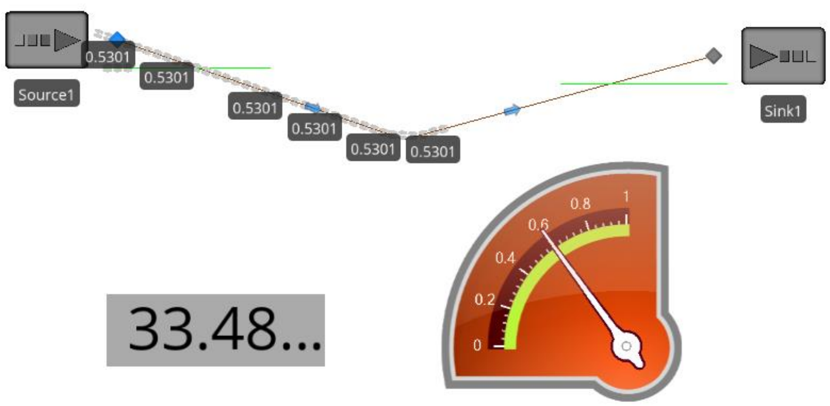
$$ConveyorGraphicSpeed = \frac{ConveyorLogicalSpeed * ConveyorGraphicalLength}{ConveyorLogicalLength}$$

$$ConveyorCapacity = \frac{ConvLogicalLength * CanPerMetre}{LocalScale}$$

As entities enter the conveyor, their length dimension is set so as to be the fraction of the conveyor occupied by the number of cans represented by the entity:

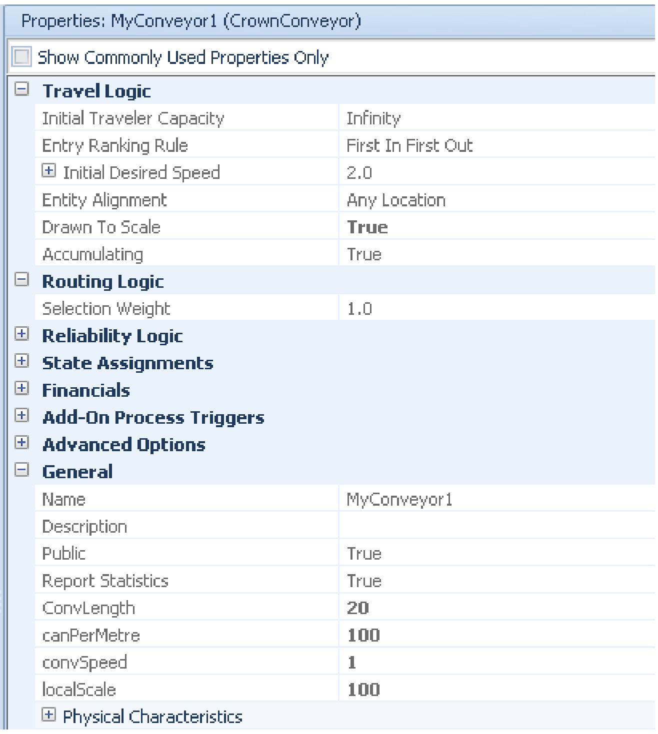
$$EntityLength = \frac{ConvGraphicalLength * LocalScale}{ConvLogicalLength * canPerMetre}$$

Graphics



Data Entry

The data in the General section can be replace by variables or data table values and therefore the conveyor properties can be driven from Excel. The extra data required for the conveyor is:



- ConvLength – The length of the conveyor in metres. This value overrides any graphical or logical length defined.
- CanPerMetre – number of cans per metre of conveyor. It is related to the can diameter and conveyor width and is easier to calculate in Excel.
- ConvSpeed – in metres per minute. This value overrides the initial desired speed.
- LocalScale – The number cans represented by an entity on the conveyor. This should include the extra scaling produced by batch nodes preceding the conveyor.

Data Output

There is no extra output above the output generated by the standard Simio Conveyor. The state of the conveyor can be gauged with the function $\frac{CrownConveyor.Contents}{Crownconveyor.CurrentTravellerCapacity}$ which tends to 1 whe the conveyor is full.