

IMPROVED CONVEYOR SYSTEM

5

This application claims priority of United States Provisional Patent Application, serial number 60/074,881, filed on February 17, 1998, the disclosure of which is incorporated herein by reference.

10

TECHNICAL FIELD

The present invention relates generally to conveyor systems, and is particularly directed to conveyor systems for controlling the spacing between packages being delivered to a sortation subsystem. The invention will be specifically disclosed in connection with a conveyor system which merges multiple lanes into a single lane for delivery to a sortation subsystem.

15

BACKGROUND OF THE INVENTION

20

Conveyor systems which combine multiple feed lanes into a single lane for sorting are well known in the art. Such systems are known to include an accumulator conveyor in line with a brake/meter conveyor on each feed conveyor lane. The accumulator conveyor typically holds packages for release either on demand to the meter conveyor or as a slug of packages. The brake/meter conveyor typically includes two belts, with the meter belt being driven while the brake belt is driven through a power take off. Such systems incorporate a ratio between the first and second belt, such that a gap between packages is created or increased as the packages are transferred across the speed ratio from the brake belt to the meter belt.

25

Some systems incorporate additional conveyors to effect the metering and delivery of packages to a merge. The typical system at present attempts to release single packages from

each lane such that single packages are interleaved, or "zippered", with single packages from other lanes as the packages are merged from various lanes.

The spacing between packages, which are typically of random length, must be kept above a minimum length in order for the sorter to direct each package to the appropriate output/discharge conveyor. The goal of such conveyor systems is to come as close to the minimum inter-package gap as possible, without dropping below. By doing this, the number of packages sorted per minute, for a given sorter speed, can be maximized. Anytime the inter-package gap between packages exceeds the minimum inter-package gap required, the space and throughput lost cannot be recovered.

A single lane typically cannot support the required throughput rate. Thus, interleaving individual packages from more than one lane is required. Multiple feed lanes are desirable because it spreads the volume requirement out over multiple lanes. Further, accumulating conveyors exhibit poor handling characteristics at high volume. The solution would seem to be a multi-lane interleaving merge where single package are interleaved from multiple lanes and merged. Unfortunately, as the number of lanes increase, the merging conveyor required becomes large, noisy and costly. To alleviate this concern, the present invention uses a conveyor system which is not as effective for interleaving individual packages, but particularly effective for interleaving slugs of packages. The present invention is particularly suited for merging slugs of packages, as contrasted to zippering individual packages. Excess gap occurs less frequently. The inter-package gap between packages in a slug is reduced to compensate for the inter-slug gap. The result is a consistent average inter-package gap delivered to the sorter without violating the minimum inter-package gap requirement.

There is a need in the art for a conveyor system which can reliably produce a consistent average inter-package gap without violating the minimum inter-package gap requirement, with the difference between the average inter-package gap and the minimum inter-package gap requirement being minimized.

SUMMARY OF THE INVENTION

It is an object of this invention to obviate the above-described problems and shortcomings of the prior art heretofore available.

It is another object of the present invention to provide a conveyor system which provides precise position control of the package to achieve the desired spacing.

5 It is yet another object of the present invention to provide a conveyor system which can accommodate multiple feed conveyor lanes while maintaining precise package position control, producing consistent average inter-package gap without violating the minimum inter-package requirement, and minimize the difference between the average inter-package gap and the minimum inter-package requirement.

10 It is another object of the present invention to provide a conveyor system which can operate at high speeds.

Additional objects, advantages and other novel features of the invention will be set forth in part in the description that follows and in part will become apparent to those skilled in the art upon examination of the following or may be learned with the practice of the invention. The
15 objects and advantages of the invention may be realized and obtained by means of the instrumentalities and combinations particularly pointed out in the description that follows.

To achieve the foregoing and other objects, and in accordance with the purposes of the present invention as described herein, there is provided a conveyor system which includes a plurality of feed conveyor lanes comprising an accumulator conveyor, a brake conveyor, a meter
20 conveyor and a staging conveyor. The meter and staging conveyor are controlled and coordinated to transfer packages to the staging conveyor in such a way so as to achieve accurately a predetermined inter-package gap between each package on the staging conveyor. Furthermore, the output speed of the meter conveyor and the input speed of the staging conveyor are controlled and coordinated so that they match at the time of transfer of a transfer region of
25 the package. When the slug of packages transferred to the staging conveyor reaches a predetermined length, the slug is then available for transfer to the spur conveyor and on to the takeaway conveyor.

5 In essence, the present invention prepares packages for high-speed sortation by staging such packages as slugs of packages on the staging belt with the required inter-package gaps therebetween. This allows the entire staging belt to be released at a high rate of speed. The slugs are then merged with an appropriate inter-slug gap on the takeaway conveyor by calculating the best moment to release a lane.

10 In accordance with one aspect of the present invention, packages are staged on the staging belt with less than the required inter-package gap therebetween. A downstream gap optimizer redistributes a trailing inter-slug gap among the inter-package gaps resulting in a consistent average inter-package gap approaching or equal to ideal, or minimum, inter-package gap.

15 In yet another aspect of the present invention, packages are staged on the staging belt very close together and the staging belt discharges the slug to the takeaway conveyor at a slower speed than that of the takeaway conveyor, resulting in an increase in the average inter-package gaps. A downstream gap optimizer redistributes the gaps, including a trailing inter-slug gap, resulting in a consistent average inter-package gap approaching or equal to ideal, or minimum, inter-package gap.

BRIEF DESCRIPTION OF THE DRAWINGS

20 The accompanying drawings incorporated in and forming a part of the specification illustrate several aspects of the present invention, and together with the description served to explain the principles of the invention in the drawings:

Figure 1 is a plan view of a conveyor system embodying the invention.

Figure 2 is a diagrammatic plan view of a portion of the conveyor system illustrated in Figure 1.

25 Figure 3 is an enlarged, diagrammatic, fragmentary side view showing the meter belt photo eyes.

Figure 4 is an enlarged, diagrammatic, fragmentary side view illustrating the various sections of a package.

Figure 5 is a legend of the symbols used in the flow diagrams of Figures 4-22.

5 Figure 6A is a flow diagram illustrating a portion of the accumulation conveyor loop logic used in the control of a conveyor system embodying the present invention.

Figure 6B is a flow diagram illustrating a part of the accumulation conveyor control loop logic used in the control of an accumulation conveyor in a conveyor system embodying the present invention.

10 Figure 6C is a flow diagram illustrating a part of the accumulation conveyor loop logic used in the control of a conveyor system embodying the present invention.

Figure 6D is an enlarged, diagrammatic, fragmentary side view showing an accumulation conveyor.

Figures 7A and B are flow diagrams illustrating the brake belt/meter belt loop logic used in the control of a conveyor system embodying the present invention.

15 Figure 8 is a flow diagram illustrating the add package to list logic used in the control of a conveyor system embodying the present invention.

Figure 9 diagrammatically and graphically illustrates the distance CD_{MAX} and the location of a package's decision point.

20 Figure 9A graphically illustrates three examples of possible velocity profiles of the trailing package.

Figures 10A and 10B are a flow diagrams illustrating the staging loop logic used in the control of a conveyor system embodying the present invention.

Figure 11A is an enlarged, diagrammatic, fragmentary side view showing a package being transferred to a staging belt.

Figure 11B is an enlarged, diagrammatic, fragmentary side view showing a package on a staging belt.

Figures 12A, 12B and 12C are graphs of three different exemplary velocity profiles for the staging belt.

5 Figures 13A and 13B are enlarged, diagrammatic, fragmentary side views showing packages on a meter belt and a staging belt.

Figure 14 is a graph of an exemplary velocity profiles of the motion and velocity of the staging belt receiving a package from the meter belt.

10 Figure 15 is a flow diagram illustrating the staging belt and meter belt logic used in the control of a conveyor system embodying the present invention.

Figure 16 is a flow diagram illustrating the run staging belt logic used in the control of a conveyor system embodying the present invention.

Figure 17 is a flow diagram illustrating the staging tick logic used in the control of a conveyor system embodying the present invention.

15 Figure 18 is a diagrammatic, fragmentary illustration of the staging belts and spur conveyors of the feed conveyor lanes, the takeaway conveyor and transport conveyor.

Figure 19 is a flow diagram illustrating the wedge plan logic used in the control of a conveyor system embodying the present invention.

20 Figure 20 is a flow diagram illustrating the chose lane logic used in the control of a conveyor system embodying the present invention.

Figure 21 is a flow diagram illustrating the compute counts logic used in the control of a conveyor system embodying the present invention.

Figures 22A and 22B are flow diagrams illustrating the compute the plan logic used in the control of a conveyor system embodying the present invention.

Figure 23 is a flow diagram illustrating the compute air priority logic used in the control of a conveyor system embodying the present invention.

Figure 24 is a flow diagram illustrating the determine minimum lane logic used in the control of a conveyor system embodying the present invention.

5 Figures 25A and 25B are flow diagrams illustrating the check for blow through logic used in the control of a conveyor system embodying the present invention.

Figure 26 is a flow diagram illustrating the decrement counts logic used in the control of a conveyor system embodying the present invention.

10 Figure 27 diagrammatically illustrates the redistribution of the inter-slug gap among the inter-package gaps of the slug

Figure 28 diagrammatically illustrates the dense packing algorithm from the staging belt to downstream of the gap optimizer.

Figures 29A and 29B are diagrammatic illustrations of various feed conveyor configurations.

15 Reference will now be made in detail to a preferred embodiment of the invention, an example of which is illustrated in the accompanying drawings.

DETAIL DESCRIPTION OF A PREFERRED EMBODIMENT

Referring now to the drawings in detail, wherein like numerals indicate the same elements throughout the views, Figure 1 shows a plan view of a conveyor system 2 which includes induction subsystem 4, conveyor line 6, sortation subsystem 8, and recirculation lane 10. Induction subsystem 4 includes a plurality of feed conveyor lanes 12a, 12b, 12c and 12d, and takeaway conveyor 14. Transport conveyor 16 is located downstream of takeaway conveyor 14. Gap optimizer 18 is located in conveyor line 6 downstream of induction subsystem 4 and upstream of sortation subsystem 8. The operation of gap optimizers are well know to persons of ordinary skill in the art in the industry, and are used to refine inter-package gaps between packages as they flow toward the sortation subsystem. A gap

20

25

optimizer typically includes a high speed brake/meter belt conveyor in which the meter belt operates at speeds above and below the downstream conveyor. The gap optimizer can at least increase inter-package gap, but preferably can also decrease gap. The gap optimizer is set up to attempt to achieve less than the average gap for the system due to tolerances of the resulting gaps. If the inter-package gaps arriving at the gap optimizer are within its tolerance range, it will not attempt to correct the gap.

The induction subsystem 4 of the present invention is intended to feed packages to high speed sortation subsystem 8 with a consistent average inter-package gap, while not violating the specified minimum inter-package gap requirement for sortation. The closer the consistent average inter-package gap and minimum inter-package gap, the more efficient induction subsystem 4. Although ideally, the maximum efficiency of induction subsystem 4 occurs when the average inter-package gap and minimum inter-package gap are equal, it may not be possible to reduce the average inter-package gap to less than two inches greater than the minimum inter-package gap requirement due to variability in package handling. Packages do not always behave in a consistent manner due to factors such as slippage, dirt, rounded corners, slippage, due to a variety of factors.

As will be described in detail, during operation of conveyor 2, slugs of packages (not shown) are delivered to takeaway conveyor 14 by feed conveyor lanes 12a-12d. The packages are conveyed by takeaway conveyor 14 to transport conveyor 16, to gap optimizer 18 and onto sortation subsystem 8 by conveyor line 6. In sortation subsystem 8, individual packages are distributed onto the appropriate one of a plurality of sortation take away lines 20a, 20b, 20c and 20d.

It is noted that in a preferred embodiment, if a particular sorter takeaway is full, the package will not divert, and the package will travel to the end of the sorter, to the recirculation line and back to the feed conveyor lane that services the recirculation line. When that feed conveyor releases packages from the recirculation line, particularly toward the end of a batch when there is a high probability that the packages on the recirculation lane were designated for the same sorter takeaway, it is important not to release them at a rate that would overfeed the same sorter takeaway. Since it can take several minutes for a package to

travel around on the recirculation lane, diverted packages at the end of the batch can delay the batch completion. Rate limiting algorithms can be employed to make sure that the sorter takeaways are not overfeed. If the recirculation lane were to become full due to too many non-diverted packages, the system would have to shut down. In terms of limiting the rate, the sum of the length of the inter-package gap plus the package length, divided the sorter speed is compared to the package length divided by the sorter takeaway speed.

Figure 2 diagrammatically illustrates a portion of induction subsystem 4 of conveyor system 2, and includes computer control 22. For simplicity, Figure 2 illustrates only two of the four feed conveyor lanes 12a-12d shown in Figure 1. Although four feed conveyor lanes are illustrated in Figure 1, it will be understood that the present invention may be practiced in a conveyor system having more than four or less than four feed conveyor lanes. Furthermore, although the feed conveyor lanes 12a-12d of conveyor system 2 are configured as a basic wedge or sawtooth type merge, the present invention may be practiced with a variety of configurations, such as shown in Figures 29A and 29B.

As shown in Figure 2, feed conveyor lane 12a comprises accumulation conveyor 28a, brake belt 30a mechanically linked to meter conveyor 40a through power takeoff 38a, staging belt 34a and spur conveyor 36a. As is illustrated, feed conveyor lane 12b comprises similar components which are numbered corresponding to the components of feed conveyor lane 12a. Other feed conveyor lanes 12c-12d are also similarly constructed, as thus will not be discussed in detail.

Feed conveyor lane 12a is fed by accumulation conveyor 28a. In a preferred embodiment, accumulation conveyor 28a is a roller conveyor which releases packages in response to a signal from computer control 22.

Immediately downstream of accumulation conveyor 28a is the brake/meter conveyor comprised of brake belt 30a and meter belt 32a. Meter belt 32a is driven by motor and motor control 24a. Brake belt 30a is driven by meter belt 32a through power takeoff 38a at a fixed speed ratio, with meter belt 32a traveling faster. As is well known to a person of ordinary skill in the art, the purpose of the speed ratio between brake belt 30a and meter belt 32a is to insure that there is an adequate inter-package gap between every pair of adjacent packages on

meter belt 32a sufficient to allow each package to be individually detected by meter belt photo eyes 40a. Although in a preferred embodiment, a fixed ratio power takeoff 38a is used to drive brake belt 30a and establish the speed ratio between brake belt 30a and meter belt 32a, the two belts may be driven in different ways in the practice of the present invention. For example, brake belt 30a could be driven by its own motor and motor control, with the ratio speeds being electronically controlled.

Meter belt position encoder 50a is connected to meter belt 32a so as to directly monitor the movement of meter belt 32a. In a preferred embodiment, meter belt position encoder 50a is a pulse encoder with one-quarter inch resolution. Signals from meter belt position encoder 50a are inputted into computer controlled 22, which, in conjunction with the detection of packages through meter belt photo eyes 40a, monitors and tracks the position and movement of packages on meter belt 32a.

In a preferred embodiment, motor and motor control 24a comprises an AC servo motor and control. The present invention, however, is not limited to such a motor and motor control, and other suitable motors and drives may be used.

In a preferred embodiment, brake belt 30a is ten feet long and meter belt 32a is 12 feet, six inches long. The brake belt must be long enough to prevent packages on accumulation conveyor 28a from pushing leading packages across brake belt 30a and onto meter belt 32a. Otherwise, an adequate inter-package gap between packages on meter belt 32a could not be guaranteed. In a preferred embodiment, accumulation conveyor 28a is The Buschman Company ACCUGLIDE accumulation conveyor. If accumulation conveyor 28a releases packages in singulation, i.e. one package at a time, there will be no pressure from packages on accumulation conveyor 28a against packages located on brake belt 30a. Thus, with singulation the brake meter belt 30a can be shorter, such as five feet. If accumulation conveyor 28a releases a slug with touching packages to brake belt 30a the 10 foot length has been found to be adequate to hold back the pressure of the slug release from the accumulation conveyor.

The length of meter belt 32a is selected based on the maximum length of the longest package anticipated to be handled by conveyor system 2, on the distance required to

decelerate a package from the meter belt maximum speed to zero, and the distance required to accelerate a package from zero to the maximum speed of the meter belt.

Staging belt 34a is located downstream of meter belt 32a so as to receive packages therefrom. Staging belt 34a is driven by motor and motor control 26a. In a preferred embodiment, motor and motor control 26a comprises an AC servo motor and control. The present invention, however, is not limited to such a motor and motor control, and other suitable motors and drives may be used.

Staging belt entrance photo eye 52a is disposed downstream of the center line of the upstream pulley (not separately shown) of staging belt 34a a distance sufficient to allow the longest expected package to be completely on staging belt 34a before the leading edge of that package reaches staging belt entrance photo eye 52a. Staging belt exit photo eye 54a is located upstream of the downstream pulley (not separately shown) of staging belt 34a a distance sufficient to allow the leading edge of a package to accelerate from zero at the location of staging belt exit photo eye 54a up to the release speed of staging belt 34a when the leading edge of the package reaches the middle of the gap between the staging belt and the spur conveyor. In a preferred embodiment, staging belt entrance photo eye 52a was located 60 inches downstream of the upstream pulley of staging belt 34a, and staging belt exit photo eye 54a was located 40 inches upstream of the downstream pulley of staging belt 34a. Staging belt entrance photo eye 52a may be used to detect the actual leading edges of respective packages located on staging belt 34a and the actual inter-package gap so as to compare where the packages actually are and what the inter-package gaps actually are to that planned by to computer control 22, thereby allowing computer control 22 to make corrections for any such induced errors. Similarly, staging belt exit photo eye 54a may be used to detect the actual location of the leading edge of the leading package of the slug of packages staged on staging belt 34a, the inter-package gaps between packages located on staging belt 34a as they are discharged therefrom, and the tail of the slug as it is discharged from staging belt 34a. In a preferred embodiment, staging belt entrance photo eye 52a and staging belt exit photo eye 54a are used only in calibrating the length of staging belt 34a.

In a preferred embodiment, staging belt 34a is 36 feet, six inches long. The selection of the length can be made dependent upon the average package length, the average inter-package gap and fill time for the staging belt, but in a preferred embodiment, the length was selected arbitrarily. Preferably, the ratios of fill time to cycle time for each of feed conveyor lanes 12a-12d of induction subsystem 4 are approximately equal. As should be apparent, once a feed conveyor lane is full and ready to discharge, the downstream feed conveyor lanes have to wait longer than the upstream feed conveyor lanes for an opportunity to release. This results, at least in part, from the physical limitations of any particular layout. As is readily apparent, the tail of any slug of packages released from an upstream feed conveyor lane, such as feed conveyor lane 12d will clear the associated spur conveyor of the upstream feed conveyor lanes first. Thus, in the embodiment illustrated, the tail of a slug of packages released from feed conveyor lane 12d will clear spur conveyor 36c first, then clear spur conveyor 36b, then clear spur conveyor 36a. Thus, feed conveyor lane 12c can release (if instructed to do so) sooner than feed conveyor lane 12a can release. In order to achieve approximately equal ratios of fill time to cycle time for all feed conveyor lanes, the respective lengths of the downstream staging belts (those closest to the entrance of the transport conveyor 16) become longer, since the wait times are a function of the physical layout. Another way of equalizing the ratio of fill time to cycle time is to reduce the size of inter-package gaps between packages staged on the downstream staging belts, allowing more packages to be placed on the same physical length of staging belt. In such a "dense pack" algorithm for populating the staging belt, the release speed of the staging belt (and corresponding spur conveyor) may be decreased, resulting in an increase in the lengths of the respective inter-package gaps between packages due to the speed difference between the takeaway conveyor speed and the release speed of the spur conveyor/staging belts.

Spur conveyor 36a is located downstream of staging belt 34a, disposed at an angle β to takeaway conveyor 14. Although in a preferred embodiment, beta equals 30 degrees, such angle is not to be considered limiting, as any suitable angle may be used depending upon the feed conveyor lane configuration desired. Spur conveyor 36a allows feed conveyor lane 12a to connect to takeaway conveyor 14 at an angle. In a preferred embodiment, spur conveyor 36a comprises a V-belt roller, although any suitable conveyor may be used in the practice of

the present invention. Spur conveyor operates at a constant speed, receiving the packages from staging belt 34a and discharging them onto takeaway conveyor 14. Spur conveyor 36a preferably does not run unless staging belt 34a is releasing packages to it, which reduces wear, noise and energy consumption.

5 Although feed conveyor lane 12a has been described as having belt conveyors 30a, 32a and 34a, a person of ordinary skill in the art understands the reference to brake belt, meter belt and staging belt to be a conventional reference to conveyors, and not limiting the types of these conveyors to belt conveyors only. Furthermore, a person of ordinary skill in
10 this art would understand the reference to brake belt, meter belt and staging belt not to be only to the actual belt itself as used in these conveyors, but in some instances to the conveyor as a whole and its function. It is noted that each of these conveyors in feeder lane 12a may be comprised of multiple conveyors serving the same functions. For example, staging belt 34a could be multiple belts linked together either mechanically or electronically.

 As packages are conveyed from brake belt 30a to meter belt 32a, an initial gap is
15 “pulled” allowing photo eyes 40a to detect individual packages. Meter belt 32a conveys the package onto staging belt 34a, which is driven by motor control and motor 26a. The velocity profiles (i.e. the speed and motion of the conveyor relative to package position) of meter belt 32a and staging belt 34a are controlled and coordinated by computer controller 22A such that the packages are discharged from meter conveyor 32a onto staging belt 34a in a manner
20 which produces controlled spacing between packages on the staging belt. Once a slug of a plurality of spaced packages has been formed on the staging belt, upon a signal from computer control 22, staging belt 34a will release the slug continuously, until the last package of the slug has been discharged from staging belt 34a, to spur conveyor 36a.

 As will be described in great detail below, the movement of meter belt 32a and
25 staging belt 34a are coordinated such that packages are discharged from meter belt 32a to staging belt 34a at an equal, fixed speed, placing each successive package onto staging belt 34a with a consistent average inter-package gap between it and the previous package. In a preferred embodiment, the maximum speed of meter belt 32a is 450 feet per minute; the maximum speed of staging belt 34a during staging, i.e., while packages are being transferred

from the meter belt to the staging belt, is equal to the maximum speed of meter belt 32a, in a preferred embodiment 350 feet per minute. When staging belt 34a is full of packages or when computer control 22 has called for the release of packages from staging 34a to takeaway conveyor 14, meter belt 32a stops discharging packages to staging belt 34a, and staging belt 34a discharges the slug of packages at a speed of 450 feet per minute. Spur conveyor 36a matches the speed of staging belt 34a when the slug of packages is being discharged thereto. In a preferred embodiment, takeaway conveyor 14 operates at a constant, fixed speed of 450 feet per minute. As is readily apparent, the vector speed of spur conveyor 36a is less than the speed of takeaway conveyor 14, resulting in a speed increase in the direction of travel of takeaway conveyor 14 as a result of the speed of spur conveyor 36a and angle beta, which increases the inter-package gap between successive packages of the slug as it is discharged onto takeaway conveyor 14. In a preferred embodiment, the speed of transport conveyor 16 matches the speed of takeaway conveyor 14. Although takeaway conveyor 14 is illustrated as a straight belt conveyor receiving packages from obliquely oriented belts, the takeaway conveyor is not limited to the disclosed configuration. A takeaway conveyor can include a combiner such as is typically used in a zippering merge.

In a preferred embodiment, the maximum acceleration rate of the meter belt and the staging belt was set at .4G's, and the maximum deceleration rate was set at .4G's. The maximum acceleration and deceleration rates are limited by the amount of rocking or movement induced in the packages. Packages having rounded bottoms or corners cannot be accelerated as quickly as square corner packages. As is also well known, packages having center of gravity and a narrow base (in the direction of travel) will also tend to rock or tip if accelerated or decelerated too quickly.

Referring to Figure 3, there is shown an enlarged, diagrammatic, fragmentary side view showing the meter belt photo eyes 40. Meter conveyor photo eyes 40a includes low detect photo eye 42a, high detect photo eye 44a, width detect photo eye 46a and batch tote detect photo eye 48a. As illustrated in Figure 3, low detect photo eye 42a and high detect photo eye 44a are mounted to a common vertical support 58a at a distance of 40 inches from the center line of upstream pulley 60a of meter belt 32a. Low detect photo eye 42a and high detect photo eye 44a are located one inch and eight inches, respectively, above conveying

surface 62a of meter belt 32a. Vertical support 58a is attached to the conveyor frame (not shown) of meter belt 32a, and also carries width detect photo eye 46a. Batch tote detect photo eye 48a is located 16 inches upstream of low detect photo eye 42a and high detect photo eye 44a, carried by vertical support 64a which is mounted to the conveyor frame (not shown).

It is preferable to have enough distance between the center line of upstream pulley 60a and low detect photo eye 42a and high detect photo eye 44a to allow the entire length of the longest expected package to be completely on meter belt 32a before the leading edge of that package reaches photo eyes 42a and 44a. This allows meter belt 32a to have complete control of the package, *i.e.*, the entire length of the package on meter belt 32a, before the leading edge of the package reaches photo eyes 42a and 44b. It is also desirable for the trailing edge of the package to be completely past photo eyes 42a and 44a before any deceleration of the package begins. Furthermore, it is preferable to be able to bring the package to a complete stop from the maximum meter belt speed and then be able to accelerate the package from zero to the maximum meter belt speed before the leading edge (or as discussed below, the leading edge of the transition section) of the package leaves the end of meter belt 32a. Any additional length is desirable but not absolutely necessary.

Width detect photo eye 46a is a proximity eye in which the sensitivity is adjusted to determine whether the packages' widths exceed a predetermined dimension. When packages are diverted in sortation subsystem 8, the trailing edge of the packages rotate backwards, reducing its trailing inter-package gap. Depending on the width of the package and the divert angle of sortation subsystem 8, this rotation may interfere with accurate sortation and diversion of packages. To avoid problems, the trailing inter-package gap is set higher for a package designated as a wide package than for a non-wide package. In a conveyor system 2 having a 20 degree divert angle for sortation subsystem 8, packages wider than 20 inches were considered "wide" packages, and the inter-package gap values were increased by about five inches.

Photo eyes 40a include tote lip detect photo eye (not shown) which is disposed to detect the leading edge of the lip of a tote. Tote lip detect photo eye is installed at a height

which matches the height of the tote lip when located on conveying surface 62a. As is well known, totes have a trapezoidal profile (when viewed from the side), with the upper tote lip being wider and longer than the base of the tote. Thus, due to the shape, the tote lip will pass the plane in which low detect photo eye 42a and high detect photo eye 44a are located before either photo eye 42a or 44a have detected the leading edge of the tote. The opposite is true at the trailing edge of the tote, in which photo eyes 42a and 44a will detect the trailing edge of the tote before the trailing edge of the tote lip has passed downstream of the vertical plane in which low detect photo eye 42a and high detect photo eye 44a are located. If the tote lip is not detected, computer system 22 will identify the tote as being shorter than it actually is, potentially resulting in problems due to insufficient inter-package gap resulting from the "overhanging"/extending tote lip on either end.

End of batch tote detect photo eye 48a is used to detect the presence of an end of batch tote on meter conveyor 32a. End of batch totes carry a reflector located on the side of the end of batch tote, typically centered height-wise and length-wise. In order to avoid false signals, low detect photo eye 42a should be located lower than the reflector on the end of batch tote, and high detect photo eye 44a should be located higher than the reflector. The height of end of batch tote detect photo eye 48a above conveying surface 62a is determined by the height of the end of batch tote. In a preferred embodiment, end of batch tote detect photo eye 48a was located four inches above conveying surface 62a. End of batch tote detect photo eye 48a is located upstream of the plane in which low detect photo eye 42a and high detect photo eye 44a are disposed a distance such that when the upper lip of the end of batch tote is detected by the tote lip photo eye, end of batch tote detect photo eye 48a detects the reflector located in the end of batch tote.

As illustrated in Figure 3, width detect photo eye 46a is located on the opposite side of vertical support 58a, spaced apart downstream from low detect photo eye 42a. Width detect photo eye 46a may be disposed in other locations, such as in the same vertical plane as low detect photo eye 42a looking down on conveying surface 62a. Width detect photoeye 46a preferably is placed at the same height as low detect 42a so that it can detect the width of short (height-wise) packages. Due to the horizontal gap between low detect photo eye 42a and width detect photo eye 46a, for adjacent packages having an inter-package gap less than

this width, at the time low detect photo eye 42a or high detect photo eye 44a detects the leading edge of a trailing package, width detect photo eye 46a may be detecting the leading package. For this reason, computer control is preferably programmed to accommodate this gap by correlating input from width detect photo eye 46a to the proper package.

5 Instead of individual photoeyes 42a, 44a and the tote lip photoeye, a light curtain could be used.

10 Figure 4 is an enlarged, diagrammatic fragmentary side view illustrating various sections of package 70. As is well known in the industry, in order to maintain precision and accuracy, when a package is transferred from one conveyor to another, the speed of the two conveyors need to match when control of the package passes from the upstream conveyor to the downstream conveyor. A conveyor has "control of a package" during transfer when the speed of that package is dictated by the speed of the controlling conveyor, with no relative motion between the package and that conveyor's conveying surface. A typical practice within the industry is to assume that the transfer of control of package occurs when the center of the package reaches the center of the gap between the upstream and downstream conveyors. Inherent in this assumption, is the assumption that the package is homogeneous, having uniform weight distribution and uniform friction characteristics throughout its length.

20 For greater precision, in one embodiment of the present invention, it is assumed that the transfer of control during transfer of a package from an upstream conveyor to a downstream conveyor occurs during the transfer of a "transfer region" from the upstream conveyor to the downstream conveyor. With reference to Figure 4, package 70 is broken into three sections: head 72, transfer region 74 and tail 76. Each section has a respective leading edges 72a, 74a and 76a and respective trailing edges 72b, 74b and 76b. As is illustrated by Figure 4, head trailing edge 72b coincides with transfer region leading edge 74a and transfer region trailing edge 74b coincides with tail leading edge 76a.

25 The lengths of the three sections may be selected based on various criteria in the practice of this invention. For example, rather than being based on a percentage of the total length of a package, the head and tail sections could be an absolute length, such as 2 inches.

It is noted that the length of head 72 and tail 76 do not have to be equal, either from a proportional or absolute basis.

There are numerous methods by which the lengths of sections 72, 74 and 76 may be determined or assigned. For example, the lengths may be calculated as a predetermined percentage of the overall length of each particular package. A variation of this would be to vary the percentages as a function of the overall length of each package, or for packages exceeding a predetermined package length. Another method is to assign a predetermined, absolute length (such as an inch) to head 72 and tail 76, leaving the length of transfer region 74 to be the remainder of the overall length of the package. Such absolute lengths for head 72 and tail 76 could be identical for every package, could be a function of the overall length of the package, or could be selected based on whether the overall length of the package exceeded a predetermined length or fell within a series of ranges. In yet another method, the center of gravity of each package could be dynamically determined based on its shift in position as the package travels across a speed ratio between the brake belt and the meter belt.

Regardless of what methodology is used to determine the respective lengths of head 72, transfer region 74 and tail 76, it is noted that any of these regions could be assigned a zero length. By way of example, transfer region 74 could be determined to have a zero length. If in such a situation, head 72 and tail 76 were assigned equal lengths, then the handling of such a package would follow the typical industry assumption that the transfer of control occurs when the center of the package reaches the center of the gap between the upstream and downstream conveyors. Alternatively, by way of example, head 72 and tail 76 may have zero length, making the length of transfer region 74 equal to the overall length of the package.

In building slugs of packages on a staging belt, the operation and coordination of each feed conveyor lane is controlled independently of the other feed conveyor lanes by respective programs resident on the main merge computer. The coordination of the release of respective feed conveyor lanes is controlled by the wedge plan logic, also resident on the main feed computer. However, the operation of a particular feed conveyor lane is ultimately affected by the operation of the other feed conveyor lanes if the plan logic calls for the release of packages from that particular feed conveyor lane.

As will be understood, the logic flow diagrams represent the logic followed by the programs which control the conveyor system. A person of ordinary skill in the art will understand that the steps illustrated do not necessarily represent a line of program code or the order in which the step is taken, also understanding that in some instances certain steps will
5 have to be taken in a particular order.

Figure 6A illustrates a portion of the accumulator conveyor loop logic used in the control of a feed conveyor lane in a conveyor system embodying the present invention. This portion of the accumulation conveyor loop logic assigns a lane priority value which is used in selecting the order in which lanes are released. If a lane is not released often enough, the
10 accumulation conveyor can back up too far and affect the upstream conveyors, such as in the picking operation or in a cross-dock operation. If an accumulation conveyor backs up too far, it can affect the upstream conveyors, such as in the picking operation or in a cross-dock operation.

The loop begins every millisecond at step 6A-1. At step 6A-2, the determination is
15 made whether the accumulation conveyor's queue 100% full photo eye has been blocked for more than a predetermined amount of time, represented by n_1 seconds. This may be accomplished in many ways, such as by the use of a counter in the logic that keeps track of how long the 100% full photo eye has been blocked. Although as described herein, photoeyes are used to determine lane status, package presence or position, it will be
20 understood that any device, signal or algorithm for monitoring such may be used.

If the 100% full photo eye has been blocked more than n_1 seconds, at step 6A-3, the program assigns the 100% full priority value to the lane. In a preferred embodiment, the lane full priority values were determined empirically, and the 100% full priority value was set at 300 inches. The priority value is used by the plan logic in determining the order in which
25 lanes are released. From there, the program ends at step 6A-9.

If the 100% full photoeye has not been blocked more than n_1 seconds, the program proceeds to step 6A-4 at which the determination is made whether the accumulation conveyor's queue 75% full photo eye has been blocked for more than a predetermined amount of time, represented by n_2 seconds. If the 75% full photo eye has been blocked more

than n_2 seconds, at step 6A-5, the program assigns the 75% full priority value to the lane. In a preferred embodiment, the 75% full priority value was set at 200 inches. From there, the program ends at step 6A-9.

5 If the 75% full photo eye has not been blocked more than n_2 seconds, the program proceeds to step 6A-6 at which the determination is made whether the accumulation conveyor's queue 50% full photoeye has been blocked for more than a predetermined amount of time, represented by n_3 seconds. If the 50% full photoeye has been blocked more than n_3 seconds, at step 6A-7, the program assigns the 50% full priority value to the lane. In a preferred embodiment, the 50% full priority value was set to 100 inches. From there, the
10 program ends at step 6A-9.

If the 50% full photoeye has not been blocked more than n_3 seconds, the program proceeds to step 6A-8 at which the program assigns the basic priority value to the lane. From there, the program ends at step 6A-9.

15 In one embodiment of the present invention, n_1 , n_2 and n_3 were about four seconds. By requiring the respective photoeye to be blocked for a period of time before taking the specified action, transitory blockages of the photoeye due to package flaps and moving packages are not counted. Additionally, the accumulation of packages back to a respective photoeye does not dictate a higher priority immediately upon blockage of the photoeye. Preferably the lane conditions will be checked by a hysteresis routine so that the lane full
20 priority assigned does not change immediately upon a photoeye becoming unblocked. In one embodiment, a two second delay was used.

25 While Figure 6A refers to 100%, 75% and 50% full photoeyes, the photoeyes may be placed at any position appropriate to aid in monitoring the condition of the accumulation conveyor. In addition, more or fewer photoeyes may be used. Furthermore, although the predetermined times have been identified with the different variables n_1 , n_2 and n_3 seconds, the time delay for each photoeye may be equal or different. The time delay may be different for different lanes. For example, a lane with an accumulation conveyor which is shorter than in other lanes (such as due to physical limitations) may need different time delays. The time delays may be different from lane to lane, or within a lane, due to the nature of the upstream

configuration. In addition, the time delays may be determined dynamically rather than being predetermined. The amount of time delay could be varied based on the conditions of the other lanes, or on the performance of the system.

5 Figure 6B illustrates a part of the accumulation control logic loop used in the control of an accumulation conveyor in a conveyor system embodying the present invention. This control operates the accumulation conveyor to prevent it from being overfed by upstream conveyors, and is independent of the logic used to control the downstream conveyors. This portion of the accumulation conveyor loop logic determines when to disable the conveyor upstream of the accumulator conveyor of a conveyor feed lane feeds packages in response to
10 the fullness of the accumulation conveyor.

The loop begins every millisecond at step 6B-1. At step 6B-2, the determination is made whether the accumulation conveyor's queue 100% full photoeye has been blocked for more than a predetermined amount of time, represented by n_4 seconds. This may be accomplished in many ways, such as by the use of a counter in the logic that keeps track of
15 how long the 100% full photoeye has been blocked.

If the 100% full photoeye has been blocked more than n_4 seconds, at step 6B-3, the program causes the upstream conveyor to be disabled and stop feeding packages to the accumulation conveyor. The program proceeds to Step 6B-6 at which the planning logic is signaled that the upstream conveyor has been disabled. From there, the program ends at step
20 6B-7.

If the 100% full photoeye has been unblocked for more than n_5 seconds, the program proceeds to step 6B-4 at which the determination is made whether the accumulation conveyor's queue 100% full photoeye has been clear for more than a predetermined amount of time, represented by n_5 seconds. If it has, the program proceed to step 6B-5 at which the
25 upstream conveyor is enabled and begins feeding packages to the accumulation conveyor. As will be readily recognized by persons skilled in the art, this n_5 time delay creates hysteresis to avoid unwanted oscillations in the enablement and disablement of the upstream conveyor.

If the 100% full photoeye has not been unblocked for more than n_5 seconds, the program proceeds to the end of the program at step 6B-6.

Although the predetermined times have been identified with the different variables n_4 and n_5 , the time delay for each step may be equal or different. The time delay may be different for different lanes. For example, a lane with an accumulation conveyor which is shorter than in other lanes (such as due to physical limitations) may need different time delays. The time delays may be different from lane to lane, or within a lane. In addition, the time delays may be determined dynamically rather than being predetermined. The amount of time delay could be varied based on the conditions of the other lanes, or on the performance of the system. In one embodiment of the present invention, n_4 was about five seconds and n_5 was about 3 seconds.

Figure 6C illustrates another part of the accumulator conveyor loop logic used in the control of feed conveyor lane in a conveyor system embodying the present invention. This portion of the accumulation conveyor loop logic determines when the accumulator conveyor of a conveyor feed lane feeds packages to the brake/meter belts to make sure that the conveyor feed lane is not overfed.

The loop begins every millisecond at step 6C-1. At step 6C-2, the determination is made whether the brake/meter belts are accepting packages from the accumulation conveyor. If so, then the program proceeds to step 6C-3, at which the accumulation conveyor is enabled (or remains enabled if already enabled), resulting in the release of packages from the accumulation conveyor to the brake belt. If the brake/meter belts are not accepting packages, the program proceeds to step 6C-4 at which the accumulation conveyor is disabled (or remains disabled if already disabled). From either step 6C-3 or 6C-4, the program proceeds to the end at step 6C-5.

In one embodiment of the present invention, the accumulation conveyor includes a drop out zone immediately upstream of the brake belt, with eight zones immediately upstream of the drop out zone. Such a conveyor is within the teachings of United States Patent Number 5,823,319, which is incorporated herein by reference. Figure 6D illustrates accumulation conveyor 28a disposed upstream of brake belt 30a of feed conveyor lane 12a.

Accumulation conveyor 28a includes a plurality of independently controlled zones, numbered 28a-x. In Figure 6D only zones seven numbered control zones, 28a-1 through 28a-7, are illustrated (control zone 28a-8 is not illustrated). Control zone 28a-1 is the drop out zone which is pulse width modulation controlled. Control zones 28a-2 through 28a-8 may be
5 operated in singulation mode or slug release mode.

When accumulation conveyor 28a is enabled, as at step 6C-3, control zone 28a-1 is active, delivering packages to brake belt 30a. Control zones 28a-2 through 28a-8 are all operated in the slug release mode.

When accumulation conveyor 28a is disabled, as at step 6C-4, control zone 28a-1 is
10 off, and control zones 28a-2 through 28a-8 are all operated in the singulation mode. Control zone 28a-1 could not prevent packages from being pushed onto the brake and meter belts if the upstream control zones 28a-2 through 28a-8 continued to operate in slug release mode. If the upstream control zones 28a-2 through 28a-8 were simply stopped when control zone 28a-1 was stopped, gaps between packages on the accumulation conveyor could be left. In
15 singulation mode, control zones 28a-2 through 28a-8 will advance packages to the end of the respective control zone before disengaging. With singulation, the "down time" of the brake/meter belts is advantageously used to remove accumulation conveyor gaps without pushing packages onto the brake/meter belts.

Turning now to Figures 7A and 7B, which illustrate the brake/meter belt loop logic
20 used in the control of a conveyor system embodying the present invention, the loop is executed once every encoder tick, as indicated at step 7-1. At step 7-2, it is determined whether the leading edge of a package is at the low detect photoeye, high detect photoeye or tote lip detect photoeye on the meter belt. To make this determination, the logic can consider whether any of these photoeyes were first broken during the last cycle of the brake meter belt
25 loop. If all of these photoeyes were unblocked during the previous execution of the loop, and any of these photoeye are blocked during this execution of the loop, then the leading edge of the package arrived at these photoeyes since the last tick of the meter belt encoder. This means that the leading edge may be downstream of these photoeyes a distance of up to the

distance traveled by the meter belt between encoder ticks (referred to herein as an encoder tick distance).

5 If it is determined that the leading edge of a package is at the photoeyes, various values are initialized at step 7-3, and data on this package is began to be collected. A file is set up in the computer's memory where the data for that package is kept until the package is discharged. This information includes the package length and "jam" length, both of which are initialized at this step as an encoder tick distance. The jam length is a variable used to determine whether a jam has occurred. If a jam occurs and packages back up to the photoeye, at least one photoeye will remain blocked until the jam is cleared. In order to
10 detect a jam, the jam length is incremented with every encoder tick that a photoeye is blocked. If the jam length exceeds a predetermined length greater than the longest expected package, the system decides that there is a jam, and shuts down the system. At step 7-3, the wide package counter is initialized as zero. At step 7-3, the package is initially identified as not being an end of batch tote. At step 7-3, this package's leading inter-package gap between
15 it and any leading package on the meter belt is initialized. Also at step 7-3, the indication is made that the meter belt is to return to V_{NORM} , as discussed below, after this package is transferred to the staging belt. From step 7-3, the program proceeds to step 7-16.

If the leading edge is not at the photoeyes, then the logic proceeds to step 7-4 at which the determination is made whether the trailing edge of the package has just cleared the
20 photoeyes. To make this determination, the logic determines whether any of these photoeyes were blocked during the last execution of the brake belt/meter belt loop, and whether all of these photoeyes are not blocked during the present execution of the loop. If the trailing edge is determined to have just cleared the photoeyes, the program proceeds to step 7-4a where it is determined whether the package length exceeds the minimum length. If the trailing edge
25 has been detected before the package is at least the minimum length, the program proceeds to step 7-4b where the program determines to ignore the entries for this package as it was not long enough to be a package. From step 7-4b, the program proceeds step 7-16. If the package is at least the minimum length, the program proceeds from step 7-4a to step 7-5, which indicates the package length is known and at which the computer prepares for the next

package, such as by advancing an index pointer in a table containing package information. The program then proceeds to step 7-16.

5 If the trailing edge of the package has not just cleared the photoeye at step 7-4, the routine proceeds to step 7-6, at which it is determined whether either the low detect or high are blocked. If neither photo eye is blocked, the program proceeds to step 7-16. If either photo eye is blocked, there is a package present, so the routine proceeds to step 7-8.

10 At step 7-8, the package length and jam length are incremented by the encoder tick distance and the package data file is updated with the information. The program then proceeds to step 7-9, at which it is determined whether the width detect photoeye 46a is detecting a wide package. If it is, the program proceeds to step 7-10 where the wide count is incremented. From step 7-10 the program proceeds to step 7-11, where it is determined whether the wide count is greater than the predetermined minimum count. In a preferred embodiment, the minimum length was about two to three inches. If the wide count exceeds the minimum, the package is designated wide at step 7-11a. If the wide sensor was not wide, 15 the program proceeds from step 7-9 to step 7-12, where the wide count is set equal to zero. Steps 7-9 through 7-12 are taken to "debounce" the sensor to avoid false signals.

20 From any of steps 7-11, 7-11a and 7-12, the program proceeds to step 7-13, at which it is determined whether end of batch tote detect photo eye 48a has detected a reflector within a predetermined distance of the leading edge of the package. The predetermined distance, x, is selected based on the length of the end of batch tote reflector. Given the upstream location of the end of batch tote detector photoeye 48a from the package detect photoeyes 42a, 44a and tote lip detect photoeye, the end of batch tote reflector will be detected about the same time that the leading edge is detected. If a reflector has not been detected by the time it should have, a distance approximately its length, an end of batch tote is not present. If an end 25 of batch tote reflector has been detected within the distance x, the package is designated an end of batch tote.

The program proceeds to step 7-14 at which it is determined whether the package length is equal to a predetermined minimum length. This is to avoid false signals from the low detect photoeye, the high detect photoeye or the tote lip photoeye, such as may happen if

a package flap interrupts any of these photoeye. If all of these photoeyes become unblocked before the package length exceeds the minimum length, then the package data file is deleted.

If at step 7-14 the package length is not equal to the minimum length, the program proceeds to step 7-14a, where it is determined whether the jam length is greater than the maximum value indicating a jam. If it is, the program proceeds to step 7-14b, where a jam is declared. If a jam is declared, the system is shut down. If the jam length is not greater than the maximum value, the program proceeds to step 7-16.

If the package length is equal to the minimum length, the logic proceeds from step 7-14 to step 7-15 at which the package is added to the list of packages on the meter belt.

Figure 8 illustrates the "add package to list" logic represented by step 7-15. In this routine, the decision point and initial release velocity, V_{REL} , of the package being added to the list is determined. In addition, in the "add package to list" logic, the release velocity of the preceding package on the same meter belt is reduced if the actual inter-package gap is too small. At step 8-1, the package is added to the list of packages on the meter belt. At step 8-2, the package's release velocity, V_{rel} , is initialized by setting it equal to the normal operating velocity of the meter belt, V_{NORM} . Next, at step 8-3, it is determined whether the package being added to the list (in this discussion the trailing package) is "too close" to the preceding package on the same meter belt (in this discussion the leading package). If the actual inter-package gap on the meter belt is too small, the previously determined V_{REL} of the leading package (which was initialized as equal to V_{NORM} when the leading package was added to the list) is reduced at step 8-4. The decision point of the trailing package is then determined at step 8-5.

Referring first to step 8-5, the goal of the system is for the meter belt to maintain control of the trailing package so that the package is not discharged to the staging belt with a too small of a leading inter-package gap. At step 8-5, the decision point is determined for the package being added to the list. The decision point is the location on the meter belt that the deceleration of the package must begin in order for the meter belt to maintain control of the package. The distance between the decision point and the center of the gap between the meter belt and the staging belt (referred to herein as the meter belt/staging belt gap center) is

designated the control distance. At step 8-5, the control distance for a particular package is set equal to the lesser of the measured control distance and the maximum control distance.

In a preferred embodiment of the present invention, the measured control distance is the actual distance available to maintain such control of the package. The measured control distance is equal to the length of the leading inter-package gap plus the length of the tail section of the leading package, or $\text{gap} + T_L$, where T_L is the length of the tail section of the leading package. T_L is included in this formula because, once the leading edge of the leading package's tail section reaches the meter belt/staging belt gap center, the meter belt may be decelerated in controlling the trailing package.

The maximum control distance for the system is the distance required to decelerate a package from V_{NORM} to zero and back up to V_{NORM} by the time that the leading edge of the package's transfer section reaches the meter belt/staging belt gap center. To do this based on each package's actual dimensions requires that the length of the package be known in order to determine the location of the leading edge of the transfer section. However, in one embodiment, when the package length reaches the minimum length (and the program reaches step 7-15), the final package length is not yet known, nor are the actual length of the head section and, concomitantly, the location of the leading edge of the transfer section. Since the actual length of the package's head section is unknown, a predetermined length, H_S , is assigned as the head section length in order to determine the location leading edge of the transfer section. H_S is the length of the head section of the shortest package expected.

The distance required to decelerate from V_{NORM} to zero is given by the formula:

$$s = \frac{1}{2} V_{\text{NORM}}^2 / a \quad \text{where } a \text{ is the deceleration rate of the meter belt}$$

The same formula applies to the distance required to accelerate from zero to V_{NORM} where a is the acceleration rate of the meter belt. Although the acceleration and deceleration rates can vary, in one embodiment, the rates are the same, which, as described above, are set at $a = .4g$, where g is the acceleration of gravity. Since the meter belt does not operate above V_{NORM} , s represents the maximum distance that will ever be required to accelerate or decelerate

between zero and V_{NORM} . This distance is designated the maximum acceleration distance, herein designated A_{MAX} .

The maximum control distance, CD_{MAX} , is given by the formula:

$$CD_{MAX} = 2 A_{MAX} - H_s$$

5 The decision point will never occur further upstream from the meter belt/staging belt gap center than this distance. The decision point of a package which is not "too close" will always occur a distance of CD_{MAX} from the meter belt/staging belt gap center. Once the acceleration and deceleration rates are set, CD_{MAX} does not have calculated more than once.

10 Figure 9 diagrammatically illustrates the distance CD_{MAX} (which is the location of a non-"too close" package's decision point). As shown in Figure 9, package leading edge 70a (which coincides with head leading edge 72a) is located at decision point DP of package 70, a distance CD_{MAX} upstream of the center of the gap 68a between meter belt 32a and staging belt 34a. In this illustration, package 70' has a length equal to the shortest expected package, and accordingly has head section 72 with a length of H_s . Since H_s is the shortest length of
15 any expected head section Package 70' represents the location of package 70 at the downstream position at which transfer section leading edge 74a' is located at the meter belt/staging belt gap center 68a, with head 72' extending beyond at least partially overlying conveying surface 66a of staging belt 34a. As had been discussed, although the speed of conveying surface 66a of staging belt 34a does not match the speed of conveying surface 62a
20 of meter belt 32a until the actual transfer region leading edge 74a' reaches meter belt/staging belt gap center 68a, staging belt 34a has no effect on package 70' since metering belt 32a has control of package 70' at least until leading edge 74a' reaches center of gap 68a.

25 Figure 9 includes a graph of the speed of leading edge 70a of package 70 versus the position of leading edge 70a. The graph represents the velocity profile of package 70 which is related to the determine the calculation of CD_{MAX} , but it does not necessarily represent the actual velocity profile that a package will follow downstream if its decision point is located CD_{MAX} upstream of the meter belt/staging belt gap center (although it may follow that profile). As illustrated, the total distance traveled by the leading edge 70a is $2 A_{MAX}$.

However, since head section 72' is allowed to extend past meter belt/staging belt gap center 68a, $CD_{MAX} = 2 A_{MAX} - H_S$, as illustrated.

Alternative methods may be used for determining a maximum control distance or the location of a package's decision point. For example, the determination could be based on leading edge 70a achieving the maximum velocity of meter belt 32a when it coincides with center of gap 68a. Other points along the length of the package, or even prior to the leading edge of the package, may be used for these determinations. Furthermore, in a preferred embodiment, the decision point is the location of the leading edge of the package at the time deceleration must begin. Other points along or relative to the length of the package, *i.e.*, a package reference point, can be used for these determinations. For example, the location of the leading edge of the transfer section at the time deceleration must begin could be used. All calculations, though, must be consistent with the package reference point used to determine the decision point.

In a preferred embodiment, the meter belt must be able to stop the leading edge of the trailing package a distance upstream of the meter belt/staging belt gap center of at least one half of the leading inter-package gap. If this cannot be done when the leading package is discharged at V_{NORM} , then the trailing package is considered "too close" to the leading package. If a package is considered "too close" to its leading package, V_{REL} of that leading package must be reduced in order to maintain control of the trailing package. Other criteria may be used to determine under what conditions the release velocity needs to be reduced and to what speed.

The evaluation of whether a package is "too close" occurs at step 8-3, before the decision point of the trailing package is determined. In one embodiment, a trailing package is considered "too close" to its leading package if the leading edge of the trailing package cannot be stopped a distance upstream from the meter belt/staging belt gap center of at least one half of the inter-package gap between the trailing package and the leading package. This is represented by the comparison:

$$\frac{1}{2} \text{ Gap} + T_L \leq A_{MAX} \quad \text{where } T_L \text{ is the length of the tail section of the leading package}$$

In this equation, A_{MAX} represents the distance required to decelerate a package from V_{NORM} to zero (if the deceleration rate is different from the acceleration rates, this would be A_{MAX} for deceleration). $\frac{1}{2} \text{ Gap} + T_L$ is referred to as the measured ramp distance, which is the available distance in which to stop the leading edge of the trailing package a distance one half inter-package gap from the meter belt/staging belt gap center. If $\frac{1}{2} \text{ Gap} + T_L$ is greater than A_{MAX} , there is enough distance between the packages to decelerate the trailing package from V_{NORM} to a stop at a distance of half the inter-package gap up from the meter belt/staging belt gap center, and the ramp distance is set equal to A_{MAX} . If the package is not "too close", then the program proceeds to step 8-5.

If $\frac{1}{2} \text{ Gap} + T_L \leq A_{MAX}$, there is not enough distance between the packages to decelerate the trailing package from V_{NORM} to a stop at a distance of half the inter-package gap up from the meter belt/staging belt gap center, and the ramp distance is set equal to $\frac{1}{2} \text{ Gap} + T_L$.

If the package is "too close", then at step 8-4, V_{REL} of the preceding package is reduce to a speed which allows the leading edge of the trailing package to be stopped half the inter-package gap upstream from the meter belt/staging belt gap center. In this instance, V_{REL} of the preceding package is given by the formula:

$$V_{REL} = (2a \times \text{ramp distance})^{1/2}$$

(It is noted that when the ramp distance is equal to A_{MAX} (the trailing package is not "too close") the above formula becomes $V_{REL} = (2a \times A_{MAX})^{1/2} = (2a \times (\frac{1}{2} V_{NORM}^2/a))^{1/2} = V_{NORM}$.) From step 8-4, the program proceeds to step 8-5.

The release velocity for the trailing package is not known at this point in time, although its assigned release velocity from step 8-2 remains at V_{NORM} . The trailing package's assigned release velocity will be remain at V_{NORM} either until its trailing package is determined to be "too close" or until it is reset by the staging loop logic. Unless and until the trailing package is completely stopped on the meter belt, the release speed it can attain varies as time progresses. The velocity that the package can attain depends on its speed and

position when the staging loop is determining the actual release velocity. Figure 9A graphically illustrates three examples of possible velocity profiles of the trailing package, starting with the release of the preceding package until the leading edge of the transfer section of the trailing package reaches the meter belt/staging belt gap center. V_{REL-1} represents the reduced release velocity of the leading package. Starting at point 78, when the trailing package is at its decision point, DP, and at V_{REL-1} , the meter belt begins deceleration toward zero at point 90. If the meter belt is instructed to release the package when it reaches point 80, the package will have enough distance to reach V_{NORM} at point 82 and continue on to point 84. If the meter belt is instructed to release the package when it reaches point 86, the package will not have enough distance to reach V_{NORM} , reaching only a lower speed at point 88. If the package completely stops at point 90, it will only reach the speed at point 92, which is lower than the speed at point 88.

Referring to Figure 7B, the program is now at step 7-16 following any of steps 7-3, 7-4, 7-6, 7-14 or 7-15. At step 7-16, the determination is made whether the first package on the list has reached its decision point. If it is, the program proceeds to step 7-17 at which the meter belt is instructed to decelerate toward zero. At step 7-18, the staging loop is signaled that there is a package on the meter belt available for release. This is done by setting the "meter belt has package" variable to true. As described below, the staging loop checks the status of the "meter belt has package" variable at step 10-12.

The program proceeds to step 7-19 from step 7-16 or 7-18. At step 7-19 it is determined whether the first package on the meter belt can be transferred to the staging belt. If the meter belt has been instructed to start, the program starts another routine at step 7-20. This routine, which runs independent of the brake belt/meter belt loop, accelerates the meter belt to V_{REL} , maintains V_{REL} while the transfer section is transferred to the staging belt, and returns the meter belt to V_{NORM} , unless the trailing package on the meter belt is "too close" (in which case the indication was made that the meter belt is not to return to V_{NORM} when the trailing package was identified as "too close", as described above) in which case the meter belt begins deceleration after the transfer section has been transferred to the staging belt.

After starting the routine at step 7-20, the package being release is removed from the list at step 7-21, but information is maintained for downstream operations. The program reaches the end of the brake belt/meter belt loop at step 7-22 from step 7-19 or 7-21.

Figures 10A and 10B illustrate the staging loop logic used in the control of a conveyor system embodying the present invention. In a preferred embodiment, the staging belt has five states, designated filling, full, forward, ready and release. Filling means that the staging belt is in the process of being staged, i.e., receiving packages from the meter belt to build a slug of packages. Filling does not mean that a package is being transferred to the staging belt, although while in the filling state packages are transferred to the staging belt. Full is a temporary state existing for an instant, which is declared once the staging belt has stopped filling. In the forward state the staging belt is moving the slug forward toward the ready position, but prior to its deceleration toward stop at the ready position. The ready state begins as soon as the staging belt begins its deceleration toward stop at the ready position. In the release state, the staging belt has started its release of the slug to the takeaway conveyor in response to a signal from the wedge planning logic.

At step 10-1, the program determines whether the staging belt encoder has generated a tick since the last execution of the staging loop. If there has been a staging belt encoder tick, then the staging tick routine is called at step 10-2, which will be described below. At steps 10-3 through 10-7, the state of the staging belt is evaluated. At step 10-3, it is determined whether the staging belt is in the filling state. If so, the program proceeds to step 10-8. If not, the program proceeds to step 10-4 where it is determined whether the staging belt is in the full state. If so, the program proceeds to step 10-5. If not, the program proceeds to step 10-6.

Starting with the staging belt filling state, at step 10-8, the transfer status of the meter belt/staging belt is evaluated. The transfer status has three possible designations: "no package", "start belts" and "completing". If the transfer status is "no package", which indicates that a package is not currently being transferred to the staging belt, the program proceeds to step 10-10. Otherwise, at step 10-9, the program calls the start staging belt and meter belt routine.

At step 10-10, it is determined whether the last package actually on the staging has been designated the last package. If it has been so designated, then there is no room on the staging belt for additional packages, and the staging belt state is set to full at step 10-11. The program proceeds to step 10-6.

5 If the last package on the staging has not been designated the last package, the program proceeds to step 10-12, where it is determined whether the meter belt has a package available to be transferred onto the staging belt. To do this, the program checks the status of the "meter belt has package" variable. If this variable is false, a package has not reached its decision point on the meter belt, there is not a package available to be transferred onto the
10 staging belt, and the program proceeds to step 10-13 where the staging belt inactivity timer is evaluated. If a package has not been available to be transferred to the staging belt for a predetermined period of time, the staging belt is considered full and its state is set to full at step 10-14. If the inactivity timer has not expired at step 10-13, or the state has been set to full at step 10-14, the program proceeds to step 10-6.

15 Steps 10-13 and 10-14 allow the staging belt to be released even though it is not completely full. It is more efficient to release a partially full staging belt if packages are not currently being delivered to it. The length of the inactivity timer is selected based on the amount of time for a package at the discharge end of the accumulation conveyor, e.g., at the drop out zone, to reach the low detect, high detect or tote lip photoeyes on the meter belt and
20 considering the efficiency of releasing the packages without undue delay. In one embodiment, the inactivity timer is about 5½ seconds. The inactivity timer may be unique for any or all of the feed conveyor lanes.

 In one embodiment, a "fast staging" timer may also be used. The fast staging mode is an optional staging method designed to improve efficiency if fewer than a predetermined
25 number of feed conveyor lanes are contributing packages to the takeaway. If a lane is in the fast staging mode, packages are staged at a higher speed than normal and the inter-package gaps are increased. If the system is operating in the fast staging mode, extra inter-package gap is added to the trailing inter-package gap as determined at step 10-15. In a preferred

embodiment, with the normal speed of the meter belt and staging belt (during staging) set at 350 feet per minute, the fast staging mode is not used.

If the "meter belt has package" variable is true, a package has reached its decision point on the meter belt and is available to be transferred onto the staging belt, and the program proceeds from step 10-12 to step 10-15. At step 10-15 the program calls the
5 determine trailing inter-package gap routine, which determines the trailing inter-package gap for the current package to be transferred.

From step 10-15, the program proceeds to step 10-16 where the space available on the staging belt is evaluated. If the length of the slug after the current package on the meter belt
10 plus the trailing inter-package gap of the preceding package which is already on the staging belt is greater than the available room on the staging belt, the current package on the meter belt cannot fit on the staging belt, and the staging belt state is set to full at step 10-17. This is accomplished by comparing the length of the slug, i.e., the distance from the leading edge of the first package of the slug (the nose of the slug) to the trailing edge of the last package on
15 the staging belt to the length of the staging belt available for a slug. This distance is the length of the staging belt from end to end minus the distance from the center of the gap between the staging belt and the spur (the staging belt/spur gap center) to the ready position of this slug. From step 10-17, the program proceeds to step 10-6.

If there is room on the staging belt, the program proceeds to step 10-18, at which the
20 space that will be available on the staging belt after the current package is transferred is evaluated. If the shortest package for which the conveyor system is programmed will not fit on the staging belt after the package currently on the meter belt is transferred to the staging belt (and considering its trailing inter-package gap as determined at step 10-15), then the current package on the meter belt is designated as the last package at step 10-19.

At step 10-20, the maximum release velocity for the current package on the meter belt
25 is determined. The maximum release velocity is the lesser of V_{NORM} and the velocity the leading edge of the transfer section can reach at the meter belt/staging belt gap center given the package's position and speed. As illustrated in Figure 9A, if the package was "too close"

to its leading package, the velocity that can be reached depends on where along the velocity profile shown in Figure 9A the package is when the staging loop logic executes step 10-20.

At step 10-21, the maximum velocity, V_{ACCEPT} , at which the staging belt can accept the current package from the meter belt is determined. V_{ACCEPT} is limited by one of three factors. First, the staging belt has a maximum speed during the transfer of a package from the meter belt V_{SBMAX} , that is equal to the maximum speed of the meter belt during the transfer, which is V_{NORM} (which in a preferred embodiment is 350 feet per minute). Second, the speed which the staging belt can attain given its current position on the velocity profile from its acceptance of the preceding, or leading, package. Third, the speed at which the current package can be transferred and brought to a stop on the staging belt at the appropriate position.

As mentioned above, the velocities of the meter belt and the staging belt match and are held constant during the transfer of the transfer section of a package. After the trailing edge of the transfer section has reached the meter belt/staging belt gap center, both belts are independently controlled to prepare to discharge or receive the next package. The staging belt follows a velocity profile designed to bring it to a stop at a selected location. This velocity profile dictates the maximum acceptance velocity for this package and the next package. The selection of the velocity profile is discussed with reference to Figures 11A and 11B.

Figures 11A and 11B diagrammatically illustrate the operation of the staging belt as it relates to the third factor which can limit V_{ACCEPT} . Figure 11A shows tail section leading edge 106a (which is also transfer section trailing edge 104b) of package 100 located at meter belt/staging belt gap center 68a. At this point, staging belt 34a can begin deceleration of package 100, and meter belt 32a can begin its velocity profile as necessary for a trailing package (not shown) on meter belt 32a. Figure 11B illustrates package 100 stopped on staging belt a distance called the stopping gap, SG, from meter belt/staging belt gap center 68a. The stopping gap represents the furthest point down stream of meter belt/staging belt gap center 68a at which package trailing edge 100b will be stopped. There are many different ways to set the stopping gap. In one embodiment of the present invention, the

stopping gap is determined based on a comparison of package 100's trailing inter-package gap, as determined at step 10-15, to the distance required by staging belt to decelerate, at its deceleration rate, from the maximum meter belt release velocity (which in a preferred embodiment is V_{NORM}) to zero and accelerate, at its acceleration rate, from zero to the maximum meter belt release velocity. The distance of travel required for the staging belt to accelerate from zero to the meter belt's maximum release velocity is referred to as the meter idle ramp. In a preferred embodiment, the staging belt's acceleration and deceleration rates are equal, both being set at .4g, and the acceleration and deceleration distances are equal. If package 100's trailing inter-package gap is greater than twice the meter idle ramp, then the stopping gap is given by the formula:

$$SG = \text{trailing gap} - \text{meter idle ramp}$$

This allows package 100 to travel as far downstream on staging belt 34a as possible yet still have enough "unused" trailing gap to accelerate from zero to the meter belt's maximum release velocity, i.e., the meter idle ramp. Accordingly, the time and distance remaining for staging belt 100 to reach the maximum transfer velocity to receive the next package is minimized.

If package 100's trailing inter-package gap is not greater than twice the meter idle ramp, the stopping gap is set equal to half of the trailing inter-package gap, G, given by the formula:

$$SG = \frac{1}{2} G$$

An alternative method is simply to set the stopping gap equal to one half of package 100's trailing inter-package gap for all conditions.

The maximum acceptance velocity if package 100's trailing inter-package gap is greater than twice the meter idle ramp, is, by definition V_{SBMAX} ($=V_{NORM}$). However, if package 100's trailing inter-package gap is not greater than twice the meter idle ramp, then V_{ACCEPT} may be less than V_{SBMAX} ($=V_{NORM}$). As shown in Figure 11A, staging belt 34a may begin decelerating as soon as tail section leading edge 106a has reached meter belt/staging

belt gap center 68a. Thus, the distance within which package 100 (and concomitantly package leading edge 100a) must be able to decelerate to zero is the length of tail section 106, T, plus the stopping gap (which is one half the trailing inter-package gap of package 100). The maximum velocity from which package 100 can be decelerated to zero within this distance is given by the formula:

$$V_{ACCEPT} = (2a (T + SG))^{1/2}$$

subject to not exceeding V_{SBMAX} ($=V_{NORM}$).

Figures 12A, 12B and 12C graphically depict three different velocity profile examples for the staging belt after the tail section leading edge has reached the meter belt/staging belt gap center. Figure 12A depicts the case where the trailing inter-package gap is greater than twice the meter idle ramp, in which case the staging belt velocity remains at its maximum velocity for receiving a package from the meter belt for a short distance before decelerating toward zero. Figure 12B depicts the case where the staging belt begins decelerating toward zero as soon as the tail section leading edge reaches the meter belt/staging belt gap center. Figure 12C depicts the case where the acceptance velocity had to be reduced and the staging belt begins decelerating toward zero as soon as the tail section leading edge reaches the meter belt/staging belt gap center. It is noted that each profile is dictated by the length of the stopping gap. It is further noted that, as an alternative to the segment of constant velocity of V_{ACCEPT} shown in Figure 12A, the staging belt velocity could be increased above V_{SBMAX} until it had to decelerate to stop at the stopping gap distance.

The second factor which can limit V_{ACCEPT} , the speed which the staging belt can attain given its current position on the velocity profile from its acceptance of the preceding, is now discussed with reference to Figures 13A and 13B. Figure 13A shows package 100 upstream of meter belt/staging belt gap center 68a. The preceding, leading package 108, is located on staging belt 34a a distance designated ATG, standing for actual trailing gap. Figure 13B shows the two packages at a later point in time with head section trailing edge 102b (which is also transfer section leading edge 104a) of package 100 at meter belt/staging belt gap center 68a. At this point, staging belt 34a has reached V_{ACCEPT} and meter belt 32a has reached V_{REL} .

Leading edge 100a trails leading package 108 by package 108's trailing gap, G_L , which was previously selected for package 108.

At any particular point in time, the length of travel available for the staging belt to reach the acceptance velocity is equal to the length of head section 102 plus G_L minus ATG .

5 The maximum acceptance velocity for package 100 is limited to by the velocity that the staging belt can reach in this distance given its current velocity, given by the formula:

$$V_{ACCEPT} = (V_{SBCURRENT}^2 + 2a(H + G_L - ATG))^{1/2}$$

Figure 14 is a graph of exemplary velocity profiles of the motion and velocity of the staging belt receiving package 100, illustrating the limitations on V_{ACCEPT} which can result depending on the staging belt's position on its velocity profile from the preceding package 108. As illustrated, the staging belt is at least at its deceleration point following receipt of package 108, which is point 110 on the graph. As discussed below, transfer of any package from the meter belt is not considered until the staging belt has at least reached its deceleration point for the preceding package. Although the staging belt velocity at this deceleration point illustrated in Figure 14 is the maximum staging belt velocity for receiving a package from the meter belt, V_{SBMAX} , it could be less, depending on the limitation on V_{ACCEPT} imposed on the preceding package when it was being transferred to the staging belt. If the staging belt is at point 112 when V_{ACCEPT} is determined, there is enough distance for the staging belt to follow the profile illustrated up to point 114 at V_{SBMAX} and to point 116 before the leading edge of the transfer section reaches the meter belt/staging belt gap center 68a. However, has if the staging belt is at point 118, which in Figure 14 is stopped, then the staging belt can only reach a velocity less than V_{SBMAX} at point 120.

Returning to Figure 10, at step 10-22, the transfer velocity for both belts, $V_{TRANSFER}$, is set equal to the lesser of V_{REL} and V_{ACCEPT} as determined at steps 10-20 and 10-21. In implementing the code, the values of V_{REL} and V_{ACCEPT} are reset to the minimum these two variables as determined at steps 10-20 and 10-21 and the appropriate data tables are updated with this value.

The program proceeds to step 10-23 at which the time, T_{MB} , required for the meter belt to deliver the transfer section leading edge of the current package to the meter belt/staging belt gap center at V_{REL} (as set in step 10-22) is determined. The time, T_{SB} , required to advance the staging belt to the appropriate position at V_{ACCEPT} (as set in step 10-22) is also determined. As will be clear to a person of ordinary skill in this art, these positions are based on when a particular point on the respective belt reaches the horizontal apex of the belt's end roller. Any difference between the positional difference between the arcuate distance of the point around a quarter of the roller and the linear travel or position of the relevant point on the package is considered insignificant relative to the package length and the actual gap between the belts. For example, in a preferred embodiment, the rollers are about 2½ inches in diameter spaced apart on a 3 inch center. Downstream, leaving an actual gap therebetween of about ½ to ¼ inch, when the crown on the rollers and the belt thickness are considered.

The distance that the staging belt has to travel until it reaches its deceleration point, the point at which it begins deceleration toward zero, is also determined whenever T_{SB} is determined. This distance is designated the deceleration count and is used to determine when to begin deceleration of the staging belt and to reset the transfer status, as described below. The total distance the staging belt must move to receive the current package and decelerate to zero is referred to its total move. The total move is equal to the sum of the leading gap of the current package, the length of the current package and the stopping gap of the current package, minus the actual trailing gap (ATG) at the point in time that the total move is determined. The portion of the total move during which the staging belt is decelerating is equal to $V_{ACCEPT}^2/2a$. The deceleration count is thus the total move minus $V_{ACCEPT}^2/2a$.

As mentioned above, the velocity the meter belt can achieve changes for a "too close" trailing package, as discussed above and shown in Figure 9A. If release of the package on the meter belt is not initiated soon after V_{REL} is set at step 10-22, the maximum release velocity that the meter belt can attain may drop, yielding an erroneous, unattainable value for V_{REL} . To account for this, at step 10-24, T_{SB} is compared to T_{MB} . If T_{SB} is greater, the meter belt will be started during the next execution of the staging loop which begins the following

millisecond, and the program proceeds to step 10-29. It should be noted that once either belt has completely stopped, its time and maximum attainable transfer velocity does not change.

If T_{MB} is less than T_{SB} , the staging belt will be started first. The program proceeds to step 10-25 at which it is determined whether the meter belt is stopped. If the meter belt is stopped, then T_{MB} and V_{REL} will not change as the meter belt waits on the staging belt to advance far enough for the meter belt to be started. However, if the meter belt is moving, V_{REL} is subject to change.

At step 10-26, a new meter belt release velocity is determined, based on the maximum velocity that the meter belt can achieve from a stopped position (see point 90 in Figure 9A). At step 10-27, the program determines whether the new release velocity determined at step 10-26 is less than the transfer velocity set at step 10-22. If it is not, the program proceeds to step 10-29. If it is, the transfer velocity is changed to the new, lower value. As at step 10-22, in implementing the code, the values of V_{REL} and V_{ACCEPT} are reset to the new release velocity and the appropriate data tables are updated with this value. The program proceeds to step 10-29.

It is noted that if at step 10-23, T_{MB} were found to be greater than T_{SB} , the meter belt will start first. It is noted that if the staging belt reaches its deceleration point before it is started, it is possible that it will not be able to reach V_{ACCEPT} upon start up. Thus, if at step 10-23, T_{MB} were found to be greater than T_{SB} , the maximum V_{ACCEPT} for the staging belt could be recalculated following steps similar to steps 10-25 through 10-28. However, such a condition will actually occur rarely, if ever, and the present embodiment does not recalculate V_{ACCEPT} .

At step 10-29 certain variables are updated. The transfer status is set to start belts. The staging belt transfer status and meter belt transfer status are set to false, and meter belt has package is set to false. It is noted that, in a preferred embodiment, if the planning is on, as discussed regarding step 19-2 below, the transfer status will not be set to start belts at step 10-29. From step 10-29, the program proceeds to step 10-6.

Returning to step 10-8, if the transfer status is not "no package" (such as "start belts" or "completing") then the program calls the start staging belt and meter belt routine at step 10-9. Figure 15 is a flow diagram illustrating the staging belt and brake/meter belt routine, identified in step 1-9, used in the control of a conveyor system embodying the present invention.

At step 15-1, the transfer status of the meter belt is evaluated to determine whether the status is "start belts". If not, the program proceeds to step 15-2, at which it is determined whether the transfer status is "completing". Regardless of the answer, the program then proceeds to the exit at step 15-19.

10 If, at step 15-1, the transfer status is determined to be "start belts" the program proceeds to step 15-3 at which the transfer status of the meter belt is evaluated. If the meter belt transfer status is true, the meter belt has been started to transfer the current package and T_{MB} is decremented at step 15-4. If the meter belt transfer status is not true, the meter belt has not started to transfer the current package, and a new T_{MB} is determined at step 15-5.

15 From steps 15-4 and 15-5, the program proceeds to step 15-6 at which the transfer status of the staging belt is evaluated. If the staging belt transfer status is true, the staging belt has been started to accept the current package and T_{SB} is decremented at step 15-7. If the staging belt transfer status is not true, the staging belt has not started to accept the current package, and a new T_{SB} is determined at step 15-8. A new deceleration count is also
20 determined at step 15-8.

From steps 15-7 or 15-8, the program proceeds to step 15-9, at which T_{DIFF} is determined. T_{DIFF} is equal to $T_{MG} - T_{SB}$. T_{DIFF} is evaluated at step 15-10, where it is determined whether T_{DIFF} is greater than zero. If T_{DIFF} is greater than zero, the meter belt should be running. At step 15-11, the program determines whether the meter belt has been
25 started by checking whether the meter belt transfer status equals true. If it has not, at step 15-12 the brake belt/meter belt loop logic is instructed to release the current package (see discussion of step 7-19 above) and the meter belt transfer status is set to true.

From steps 15-10, 15-11 or 15-12, the program proceeds to step 15-13, where it is determined whether T_{DIFF} is less than zero. If T_{DIFF} is greater than zero, the meter belt should be running. At step 15-14, the program determines whether the staging belt has been started by determining whether the staging belt transfer status equals true. If it has not, at step 15-15
5 the program instructs the run staging belt loop to start. At step 15-16, the staging belt transfer status is set to true and certain variables are updated. If the current package is the first package, the exit count (the distance from the nose of the slug to the end of the staging belt) is updated to the new distance. The slug length, which is the distance between the nose of the slug and the trailing edge of the last package of the slug, is increased by the length of the
10 current package and its leading inter-package gap. The tail count variable is set equal to the length of the tail section of the current package. These distances are all measured in staging belt ticks. As a person of ordinary skill in the art will readily know, the distances, either in actual distance or in ticks from an encoder, and time, whether elapsed time or real time, are interrelated and may be used in the control of a conveyor system embodying the present
15 invention.

From steps 15-13, 15-14 or 15-16, the program proceeds to step 15-17 where it is determined whether both the meter belt and the staging belt have been started. If both belts have started, the transfer status is set to completing, the package is counted and slug and package information is updated at step 15-18.

20 From steps 15-17 or 15-18, the program proceeds to step 15-19, from where the routine returns to the staging loop as the completion of step 10-9 and the staging loop proceeds to step 10-6.

Figure 16 illustrates the run staging belt logic which is started by step 15-15. It is called anytime the staging belt is started, such as by steps 10-5 and 10-7, as described below.
25 At step 16-1, the program is provided the length of motion of the staging belt, the acceptance velocity and the deceleration count. The length of motion, which when called by step 15-15 is the total move, as discussed above. At step 16-2, the program sets the variable "deceleration valid" to true. This variable is used to determine whether to decrement the deceleration count and begin deceleration, as described below in conjunction with the staging

tick program, Figure 17. At step 16-3, the velocity profile of the motion is determined based on the staging belt's acceleration rate, the acceptance velocity and the staging belt's deceleration rate. At step 16-4, the program causes the staging belt's motor control to control the staging belt's motor to follow the velocity profile determined at step 16-3.

5 Although the velocity profiles depicted herein include straight line acceleration deceleration and constant speed movement therebetween, other profiles may be used without departing from the principles of the present invention. For example, the velocity profile could avoid any discontinuities in the profile as are found at the intersection of the straight line accelerations or decelerations with the constant velocity segment. The profile could be
10 calculated to avoid any jerking of the packages using higher order equations for the motion.

 Returning to Figure 10, once the staging belt cannot hold any additional packages, the state of the staging belt is set to full, as at steps 10-11, 10-13 or 10-17. If the staging belt state is full, the staging belt is ready for the nose of the slug to be advanced to the ready position. During the next execution of the staging loop, the program proceeds to step 10-5
15 where information about the movement of the staging belt required to advance to nose of the slug to the ready position is determined, the run staging belt loop is instructed to start and the staging belt state is set to forward.

 The run staging belt loop needs information about the staging belt velocity, the length of motion of the staging belt and the deceleration count (the distance the staging belt has to travel before beginning to decelerate toward zero so as to place the nose of the slug at the ready position). Although the velocities could be different, in a preferred embodiment, the staging belt advances the slug to the ready position at the same velocity at which it ultimately releases the slug to the spur, $V_{SBRELEASE}$. In a preferred embodiment, the velocity is 450 feet
20 per minute.

25 The length of the staging belt motion is the distance from the nose of the slug to the ready position. The ready position is the position of the nose of the slug upstream of the staging belt exit from which the staging belt can accelerate the slug from zero up to the release velocity of the staging belt, $V_{SBRELEASE}$, by the time that the nose of the slug reaches the staging belt/spur gap center. The distance, RP, is given by the formula:

$$RP = V_{SBRELEASE}^2/2a$$

where RP is the distance from the ready position to the staging belt/spur gap center and a is the acceleration rate of the staging belt

5 The length of the staging belt motion is thus the distance of the nose of the slug minus RP. In order to stop the nose of the slug at the ready position, the staging belt must began to decelerate the nose of the slug upstream of the ready position. This distance is $V_{SBRELEASE}^2/2a$. Thus, the deceleration count is the length of the staging belt motion minus $V_{SBRELEASE}^2/2a$.

10 The run staging belt routine, Figure 16, functions as described above, using the information determined at step 10-5. From step 10-5, the staging loop proceeds to step 10-6.

15 If the staging belt state is not full at step 10-4, or from any of steps 10-5, 10-9, 10-11, 10-13, 10-17 and 10-29 the program proceeds to step 10-6 at which it is determined whether the wedge plan has signaled (step 22-18 or 26-20) the staging belt to release the slug to the spur. If not, the staging loop proceeds to step 10-30. If the staging belt has been signaled to release, the program proceeds to step 10-7, at which information about the movement of the staging belt required to release the slug to the spur is determined, the run staging belt loop is instructed to start and the staging belt state is set to release.

20 The run staging belt loop needs information about the staging belt velocity, the length of motion of the staging belt and the deceleration count (the distance the staging belt has to travel before beginning to decelerate toward zero). As discussed above, the staging belt releases the slug to the spur at $V_{SBRELEASE}$, which in a preferred embodiment, is 450 feet per minute.

25 The length of the staging belt motion is the distance required to deliver the last package of the slug to the spur and decelerate to zero. The distance required to decelerate the staging belt from $V_{SBRELEASE}$ to zero is $V_{SBRELEASE}^2/2a$ where a is the deceleration rate of the staging belt. The end of the last package of the slug is a distance of the slug length plus the exit count. The staging belt can begin decelerating as soon as the tail section leading edge of the last package of the slug reaches the staging belt/spur gap center. Thus, the length of the staging belt motion is the exit count plus slug length minus the length of the tail section of the

last package plus $V_{SBRELEASE}^2/2a$. The deceleration count is the length of the staging belt motion minus $V_{SBRELEASE}^2/2a$, or simply the exit count plus slug length minus the length of the tail section of the last package.

5 The run staging belt routine, Figure 16, functions as described above, using the information determined at step 10-7. From step 10-7, the staging loop proceeds to step 10-30. Beginning at step 10-30, the staging loop determines certain conditions relevant to the execution of the wedge plan, described below. At step 10-30, it is determined whether the staging belt state is filling. If not, the program proceeds to step 10-31, where it is determined whether the staging belt state is full. If not, the program proceeds to step 10-32, where it is
10 determined whether the staging belt state is forward. If not, the program proceeds to step 10-41, which will be discussed below.

If at step 10-30, the staging belt state is filling, the program proceeds to step 10-33 where the transfer status of the staging belt is evaluated. If the transfer status is not "no package" (meaning that the meter belt or the staging belt have begun to transfer or receive a
15 package), the program proceeds to step 10-41, at which the lane is designated as not ok to release and not ok to commit.

If the transfer status is "no package", the program proceeds to step 10-34, where it is determined whether the lane is in blow through mode. If the lane is in blow through mode, the program proceeds to step 10-35. If the lane is not in blow through mode, the program
20 proceeds to step 10-36.

At step 10-35, it is determined whether the staging belt has at least one package on it. If there is not at least one package on the staging belt, the program proceeds to step 10-36. If there is at least one package on the staging belt, the program proceeds to step 10-37.

At step 10-36, it is determined whether the length of the slug on the staging belt is
25 long enough for early release. If it is not, the program proceeds step 10-41, which is discussed above.

The program reaches step 10-37 from any of steps 10-31, 10-32, 10-35 and 10-36. At step 10-37, the lane is designated as ok to release. At step 10-38, the program then determines whether the staging belt is stopped. If the staging belt is stopped, the program proceeds to step 10-39 where the lane is designated as ok to commit. If the staging belt is not
5 stopped, the lane is designated as not ok to commit. The staging loop ends at step 10-42, which is reached from any of steps 10-39, 10-40 and 10-41.

Returning to step 10-1, every time there has been a staging belt encoder tick since the last execution of the staging loop, a call is made at step 10-2 to the staging tick routine. Figure 17 illustrates the staging tick logic. At step 17-1, the exit count, which was originally
10 set at step 15-16, is decremented by the number of ticks which have occurred since the last execution of the staging loop. At step 17-2, it is determined whether the exit count is equal to zero, which would indicate that the nose of the slug is at the staging belt/spur gap center. If so, then the wedge plan logic is notified at step 17-3.

The program proceeds to step 17-4 from step 17-2 or step 17-3. At step 17-4, the
15 trailing gap is incremented. At step 17-5, the determination is made whether the tail count is greater than zero. If the tail count is positive, the trailing edge of a package being transferred has not reached the meter belt/staging belt gap center and there is not a trailing gap yet. If the tail count is positive, it is decremented at step 17-6 and is evaluated at step 17-7 to determine whether it is now zero. If the tail count is zero, the end of the package is at the meter
20 belt/staging belt gap center and the trailing gap has just started. If so, the trailing gap is set equal to zero at step 17-8.

The program proceeds to step 17-9 from steps 17-5, 17-7 or 17-8. At step 17-9, it is determined whether the "deceleration valid" variable is true. "Deceleration valid" is set to true whenever the run staging belt routine, Figure 16, is executed. If "deceleration valid" is
25 true, the staging belt has begun a velocity profile but has not begun to decelerate. If "deceleration valid" is true, the program proceeds to step 17-10, at which the deceleration count (the distance the staging belt still has to travel before it begins to decelerate) is decremented. At step 17-11, it is determined whether the deceleration count is zero. If it is,

at step 17-12, "deceleration valid" is set to false, and "run" is set to false. When "run" is false, the staging belt motor control decelerates the staging belt motor toward zero.

5 At step 17-13, the transfer status is set to "no package". If the staging belt state is filling, once staging belt deceleration begins, the staging loop will consider the next package on the meter belt once it has reached its decision point. This is accomplished by setting the transfer status to "no package", which, when the staging belt state is filling, leads to steps 10-10 through 10-29.

10 At step 17-14, it is determined whether the staging belt state is releasing. If it is, since the staging belt has reached its deceleration point, a new slug can begin to be built. Thus, at step 17-15, the staging belt state is set to filling.

The program proceeds to step 17-16 from step 17-14 or 17-15, where it is determined whether the staging belt state is forward. If it is, since the staging belt has reached its deceleration point and is decelerating the nose of the slug toward the ready position, the staging belt state is set to ready at step 17-17.

15 The program proceeds to exit at step 17-18 from steps 17-9, 17-11, 17-16 or 17-17.

20 Note that if the current package was designated as the "last package" at step 10-19, after reaching its deceleration point while filling, the staging belt will not decelerate from V_{ACCEPT} very long before it accelerates toward $V_{SBRELEASE}$ and advances the slug toward the ready position. This is because when the deceleration point is reached, the transfer status is set to "no package" at step 17-13. In the next execution of the staging loop, the staging belt state will be changed to full at step 10-11. In the subsequent execution of the staging loop, step 10-5 will start the run staging routine, which will immediately begin to accelerate the staging belt up to $V_{SBRELEASE}$ from the staging belt's then current velocity. The total elapsed time will be less than 3 milliseconds between the arrival of the tail section leading edge of the current package and the beginning of the acceleration of the staging belt toward $V_{SBRELEASE}$.

25

However, if the current package was not designated as the last package at step 10-17, the deceleration of the staging belt from V_{ACCEPT} toward zero will not be interrupted until

(and unless) it is determined at step 10-16 that the next available package on the meter belt cannot fit on the staging belt.

The logic used to coordinate the release of all of the feed conveyor lanes is referred to as the "wedge plan". These routines of the "wedge plan" plan the sequence and timing of the release of slugs from the staging belts to the associated spur conveyor and on to the takeaway conveyor.

Figure 18 is a diagrammatic, fragmentary illustration of the staging belts and spur conveyors of the feed conveyor lanes, the takeaway conveyor and transport conveyor in accordance with the descriptions above. Six lanes, identified as lanes A-F are shown, each having respective staging belts 34a-34f and spur conveyors 36a-36f. (For computer programming convenience, the lanes are assigned numbers not letters.) All of the lanes flow into the takeaway conveyor 14, which discharges to the transport conveyor 16. Takeaway 14 includes takeaway exit photoeye 122 near the takeaway exit 124. Photoeye 122 is used as a reference point to determine various distances or times necessary to plan the sequence and timing of the release of slugs onto takeaway conveyor 14. Preferably, photoeye 122 is located downstream of the most downstream spur conveyor 36a a distance of at least the maximum package length. This provides a distance in which the longest package will be completely on takeaway conveyor 14 and have some time to straighten from the transition onto takeaway conveyor 14. However, photoeye 122 also should not be too far downstream of spur conveyor 34a so as to limit the distance which a package position can shift. As is well known, light packages have a greater tendency to shift position.

Also diagrammatically illustrated in Figure 18 is a representative slug 126 on lane A (for simplicity only lane A is illustrated with a slug). Slug has nose 126a, which is the leading edge of the leading package of the slug, and tail 126b, which is the trailing edge of the last package in the slug. Whether or not the staging belt is completely full, as long as there is one package on the staging belt, there is a slug nose and tail. Slug nose 126a is shown located at the slug's ready point 128.

There are various distances or lengths, whether calculated in time, distance or encoder ticks, which are relevant to the operation of the wedge plan. Each lane has a respective

distance from the staging belt/spur gap center to the takeaway exit photoeye 122, which is called the spur offset. Each slug has a nose of the slug to the staging belt/spur gap center distance and slug length. Since the slug may be undergoing an acceleration when the nose is advanced to the staging belt/spur gap center, this distance is typically measured in
5 milliseconds.

Some distances are determined by the wedge plan during execution. These include the slug nose to the takeaway exit photoeye, the slug tail to spur (staging belt/spur gap center), the slug tail to takeaway exit photoeye, and the end of the inter-slug gap to takeaway exit photoeye.

10 Since the planning is done based on the slug's position on the takeaway conveyor, the distances and times are expressed in terms of ticks of the takeaway encoder (takeaway ticks), which in some cases requires simple mathematical conversion and calibration. (In a preferred embodiment, calibration was necessary to correlate distances and time on the staging belt and spur to takeaway ticks.) The takeaway ticks represented about $\frac{3}{4}$ inches per tick at 450 feet
15 per minutes.

Figure 19 illustrates the logic used in the wedge plan to coordinate the release of slugs from the feed conveyor lanes. In a preferred embodiment, the wedge plan begins execution at step 19-1 every 20 milliseconds, although the present invention may be practiced with the wedge plan executing at a different frequency. In the present case, the takeaway
20 conveyor moved only 1.8 inches in 20 millisecond at 450 feet per minute.

At step 19-2, the planning is set to on, which, in a preferred embodiment, prevents staging belts from adding packages from the associated meter belt to a partial slug. In a preferred embodiment, while a package is being transferred to a staging belt, that lane is not available to be considered for release. It is possible to allow such a lane to be considered for
25 release as long as that lane's velocity profile during transfer is accounted for. At step 19-3, the wedge plan calls the check for blow through routine. At step 19-4, the wedge plan calls the choose lanes routine. At step 19-5, the wedge plan calls the compute the plan routine. At step 19-6, the planning is set to off, which allows staging belts to resume adding packages until the next execution of the wedge plan. Step 19-7 is the end of the wedge plan execution.

Figure 20 illustrates the choose lanes logic used in an embodiment of the present invention. This routine identifies lanes which are "plannable" meaning that the lane can be considered when computing the plan at step 19-5. With each execution of the wedge plan, a new plan is developed, as will be described below. Thus, at step 20-1, the plannable status of all lanes is cleared. This is done by clearing the plannable flags which were set during the preceding execution of the wedge plan. Commensurate with this, the number of plannable lanes is set to zero at step 20-2, clearing this variable from the previous execution of the wedge plan.

At step 20-3, the loop for considering each lane is begun. In this loop, it is determined at step 20-4 whether lockout is on. If any lane is in lockout, lockout is on. If lockout is on, only lanes which are in lockout can be considered when computing the plan. If lockout is on, the program proceeds to step 20-5 to determine whether the specific lane under consideration is in lockout. If this lane is not in lockout, then it cannot be released and is not plannable, and the plan proceeds to step 20-10.

If the lane is in lockout, or if lockout was not on, the program proceeds to step 20-6, where it is determined whether this lane is ok to release. A lane is ok to release if the lane is not in maintenance mode, the lane is enabled, the lane is "ready to release", the lane status is complete, the lane is not in the hold for batching mode and the lane is not locked out.

In a preferred embodiment, lanes may be placed in maintenance mode by the operator for various reasons, such as to service components of the lane. If a lane is in maintenance mode, it will not (or should not) have any packages on the staging belt which are to be released. The system will continue to function with any lanes which are not in maintenance mode. This is accomplished by excluding from consideration any lanes which are identified as being in maintenance mode, such as by setting a flag or similar programming technique. The status is tested as the program routines execute.

Continuing with the determination of whether a lane is ok to release at step 20-6, if a lane is not enabled, or disabled, by the operator, such as in the event of a jam, the lane is not available to be considered in computing the plan. Whether a lane is "ready to release" depends on the activity of the staging belt. As long as a package is not being transferred from

the meter belt to the staging belt, the lane is considered ready to release. The lane status is complete if the lane is not releasing and has not been designated as committed. If a lane has an end of batch tote, the lane cannot be released until either all lanes are in the batching mode or the operator cancels the hold for end of batch, permitting the lane to be considered in the plan.

If it is not ok to release this lane, the lane is not plannable, and the plan proceeds to step 20-10. If it is ok to release this lane, the lane flag is set to plannable at step 20-7, the number of plannable lanes is incremented at step 20-8 and the compute counts routine is called at step 20-9.

Once the program reaches step 20-10, it returns to step 20-3 and repeats the loop for each lane. After every lane has been considered, the program returns to step 19-5.

Figure 21 illustrates the compute counts logic used in an embodiment of the present invention. This routine is called by step 20-9 and determines certain distances or times, all of which are converted to/expressed in takeaway counts, also referred to as takeaway ticks, relative to the slug on that lane for each plannable lane. All of this information is stored in a data file relative to that lane.

At step 21-1, the slug length count is determined. The slug length count is the length of the slug on the lane, from the slug nose to the slug tail (the trailing edge of the last package of the slug).

At step 21-2, the counts to release is determined. The counts to release is the time required for the nose of the slug to reach the spur (i.e., the staging belt/spur gap center) converted to/expressed in takeaway ticks. Since the time required for the nose of the slug to reach the spur varies not only with distance but also in dependence on the velocity of the staging belt at the time the lane is considered (which may require a speed change to advance the nose), time is used in a preferred embodiment for the initial determination of counts to release. This time is then converted to takeaway ticks.

At step 21-3, the front to exit count is determined. The front to exit count is the number of counts required for the slug nose to reach the takeaway exit photoeye. This is equal to the "counts to release" plus the spur offset minus the spur overshoot count.

At step 21-4, the back to exit count is determined. The back to exit count is the number of counts required for the slug tail to reach the takeaway exit photoeye. This is equal to the slug length count plus the front to exit count.

At step 21-5, the back to spur count is determined. The back to spur count is the number of counts required for the slug tail to reach the spur. This is equal to the counts to release plus the slug length count.

At step 21-6, the spur overshoot count is determined. Depending on the number and mass of packages on a staging belt, the staging belt may overshoot the ready position. Since the ready position is the location from which the slug nose can just be accelerated up to release velocity when it reaches the spur entrance (i.e., the staging belt/spur gap center), the overshoot is the distance past the spur entrance the slug nose will reach the release velocity. Since overshoot places the slug nose closer to the spur entrance than desired, overshoot has the same effect as shrinking the spur length and therefore, shrinking the spur offset. Overshoot can be determined by comparing the actual nose position to the ready position. As mentioned above, overshoot is used in determining the front to exit count. Thus, it will be obvious to a person of ordinary skill in the art that the overshoot count must be determined before the front to exit count is determined. The overshoot count is zero if there is not overshoot.

At step 21-7, it is determined whether the staging belt for this lane has "garbage". A staging belt has garbage if the contents are unknown, i.e., complete information about the slug is not known. As will be discussed below, during start up, following jams, and similar events, the exact number and location of packages on the staging belt are not reliably known. In such a circumstance, to make sure that inadequate gap is not introduced into the system by the release of a lane whose contents are not known, the entire length of the staging belt is assumed to be full of packages touching each other and extra inter-slug gap is introduced for the gap optimizer to work with. This assumption means that the nose of this "garbage" slug

cannot achieve release velocity before the nose reaches the spur entrance. The staging loop provides the indication of whether a lane has garbage.

5 If the staging belt is designated as having garbage, at step 21-8 the inter-slug gap is set equal to a value selected to provide extra inter-slug gap to accommodate a full lane. In a preferred embodiment, the extra inter-slug gap was user settable, and was set at 100 inches. It is selected to provide the gap optimizer enough room to recover if it is overfed by garbage.

If the staging belt is not designated as having garbage, at step 21-9, the inter-slug gap is set equal to the inter-slug gap determined by the determine trailing gap routine, which determines the trailing gap at step 10-15.

10 The program proceeds to step 21-10 from step 21-8 or 21-9, where the gap to exit count is determined. The gap to exit count is the number of counts required for the tail of the trailing inter-slug gap to reach the takeaway exit photoeye. This is equal to the back to exit count plus the trailing inter-slug gap as set at step 21-8 or 21-9. Following step 21-10, the program returns to the choose lane routine, proceeding to step 20-10.

15 Figures 22A and 22B illustrate the compute the plan routine used in an embodiment of the present invention. This routine is called by step 19-5. The "plan" establishes, through an iterative process, the sequence and time that lanes are released to the takeaway conveyor.

20 There are many different ways to express the structure of the plan in a computer. In a preferred embodiment, the plan is maintained as a table in a spread sheet with lanes entered in the order that they either will be released or are being considered for release. Each entry in the plan includes an identification of the lane, its status in the plan, its back to exit count, its gap to exit count and its wait to release count. The table includes a pointer which identifies the first lane in the plan and a pointer which identifies the last lane in the plan.

25 In computing the plan, the routine attempts to minimize the "air" which will be created between slugs on the takeaway conveyor. Air is the distance, in takeaway ticks, between the back of the leading slug's inter-slug gap and the nose of the trailing slug, and represents lost space and lower efficiency. During the compute the plan routine, each lane that is flagged as

plannable is sequentially examined to determine which one will result in the minimum air priority if it were to follow the last lane in the plan. Once the minimum lane out of all of the plannable lanes is identified, the remaining plannable lanes are sequentially reexamined to determine which one would result in the minimum air priority if it were to follow the just selected lane. This process is repeated until all plannable lanes have been considered. If a lane can be released as it is inserted into the plan, it will be, and any "planned" lanes ahead of it will become "committed". Only lanes which are releasing or are committed remain in the plan. All lanes put into the plan by the compute the plan routine which are not committed or releasing by the time the routine is completed are deleted from the plan before the next execution of the wedge plan.

At step 22-1, the last gap to exit count is determined. This value is taken from the gap to exit count of the plan's last entry. If there is not a last entry in the plan, then the last gap to exit count is set equal to zero. Step 22-2 is a counter to determine when all of the plannable lanes have been considered by this routine. The number of plannable lanes is determined initially at step 20-8. If the number of plannable lanes is not greater than zero, all plannable lanes have been considered and the routine is done, proceeding to step 22-24.

At step 22-3, the minimum lane number is initialized at zero. This variable is used to keep track of whether a lane with minimum air has been identified by the computer the plan routine.

At step 22-4, additional inter-slug gap for the last entry in the plan is determined. This determination is done for the last entry in the plan being computed each time step 22-4 is executed as the compute the plan routine completes its program loops.

Additional inter-slug gap is extra gap in addition to the trailing inter-slug gap set at step 21-8 or 21-9. In some instances, additional air needs to be inserted between slugs. This is done by adding additional inter-slug gap. For example, an extra gap may be necessary to accommodate various needs of the operator. One such need is to insert packages which were not read by the scanner. As is well known, packages which are not read may be diverted in the sortation subsystem to the one of the sortation takeaway line. Although frequently such no-reads are put into the system at the end of a batch, an operator may wish to get the non-

read package immediately back onto the conveyor, which requires additional inter-slug gaps to be added. The system may allow the operator to select an amount of additional inter-slug gap and when to add it, such as upon request. Or the system may insert additional inter-slug gap on a periodic basis. In a preferred embodiment, every slug had an additional inter-slug gap of one hundred inches.

Additional inter-slug gap can be added to compensate for the system's performance. For example, if the gap optimizer is being overfed, the gap optimizer can signal the wedge plan and additional inter-slug gap inserted on a reactive basis. It is noted that, due to the gap optimizer's downstream position, there will be a time lag before such additional inter-slug gap can relieve an overfeed condition at the gap optimizer.

As another example, if the slug has grown in number of packages, such as happens when two packages reach the meter belt photoeye side by side, and the skew rollers have put the two packages in single file, the gap optimizer will be overfed and the wedge plan can be signaled for additional inter-slug gap. Preferably, any slug growth can be detected at the takeaway exit photoeye and the wedge plan can include additional inter-slug gap nearer to the problem.

Compensating inter-slug gap could also be added on a predictive basis, such as based on the performance history of performance. Such additional inter-slug gaps could be added on a per lane basis. Additional inter-slug gap could be added to account for problems due to unknown causes.

In some instances, the slug length grows as packages are jostled by the transfer from the spur conveyor to the takeaway conveyor. Additional inter-slug gap could be added on a predictive basis if, for example, the amount of growth can be correlated to the package size, lane, release velocity, etc. Alternatively, a downstream photoeye, such as takeaway exit photoeye, could be used to monitor slug length growth, and additional inter-slug gap added to the next slug.

Steps should be taken to prevent any additional inter-slug gap added on a predictive basis from being considered by any reactive methodology. Otherwise, the desired additional inter-slug gap may be removed by the reactive scheme.

Step 22-5 is the beginning of the loop during which each plannable lane is considered. At step 22-6, any compensating inter-slug gap for the lane being considered is determined. In a preferred embodiment, the compensating inter-slug gap was zero for all lanes.

At step 22-7, the air value for this lane is determined, given by the formula:

$$\text{Air} = \text{this lane's front to exit count} - (\text{last gap to exit count} + \text{compensation gap} + \text{additional gap})$$

The additional gap is from step 22-4 while the compensation gap is from step 22-6. As can be seen from this equation, while the extra inter-slug gap is not used to increase the trailing inter-slug gap set at steps 21-8 or 21-9, it can affect the release of the next slug.

At step 22-8, it is determined whether this air value is less than zero. If the air is less than zero, this lane will have to wait to be released if it is added to the plan. A non-committable lane or "early planning" lane cannot be added to the plan if its air is negative because their front to exit count is dynamically changing.

At step 22-9 it is determined whether this lane is ok to commit, as determined staging loop. If it is not ok to commit, the lane cannot be considered for the plan, its lane flag is set to not plannable at step 22-11, and the number of plannable lanes is decremented. The program proceeds to step 22-14, skipping this lane during the rest of the execution of the compute the plan routine.

If the lane is ok to commit, the program proceeds to step 22-10, where it is determined whether this lane is an early planning lane as determined by the staging loop. If it is an early planning lane, its lane flag is set to not plannable at step 22-11. If it is not an early planning lane, the program proceeds to step 22-12, which calls the compute air priority routine.

Figure 23 illustrates the compute air priority routine used in an embodiment of the present invention. This routine determines the "air priority" of the lane under consideration

for use in selecting which lane should be next added the plan during this execution of the compute the plan routine. The air priority routine in combination with the determine minimum lane routine described below identifies the best next lane to add to the plan based on priorities selected to optimize system performance.

5 At step 23-1, the air priority for this lane is initialized by setting it equal to this lane's air from step 22-7. At step 23-2, it is determined whether this lane is in lockout. If it is not, then at step 23-3, its lane full priority value, which was set by the accumulation conveyor loop at step 6A-3, step 6A-5 or step 6A-7, as converted in takeaway ticks, is subtracted from
10 its air priority. As discussed above, the lane full priority value is based on the fullness of the lane. The lane full priority value represents how much to reduce its air priority. As a result, a full lane can release with more actual air than a less full lane with less actual air. This additional air is the trade off for managing the fullness of the lanes.

 If the lane is in lockout, the program proceeds to step 23-4, where it is determined whether this lane is an early planning lane. If it is, then at step 23-5, the early release penalty
15 is added to its air priority. As discussed above, lanes which are released are subjected to a penalty. This penalty is to make sure that the full lanes have a priority over partial lanes. In a preferred embodiment, the early release penalty was set at 50 inches, in takeaway ticks.

 The compute air priority routine ends following steps 23-4 or 23-5, and returns to step 22-13 of the compute the plan routine on Figure 22B. At step 22-13, the determine minimum
20 lane routine is called.

 Figure 24 illustrates the determine minimum lane routine used in an embodiment of the present invention. This routine determines which plannable lane has the lowest air priority value as the loop from step 22-5 to step 22-14 is repeatedly executed. At step 24-1, it is determined whether the minimum lane number is zero. The minimum lane number was
25 initialized to zero at step 22-3. If it is zero, it is the first time through the routine and the program proceeds to step 24-2 where the minimum lane number is set equal to the lane number of the current lane under consideration.

If the minimum lane number is not zero, the program proceeds to step 23-4 where it is determined whether the air priority of the designated minimum lane is greater than the air priority of the current lane under consideration. If it is, the program proceeds to step 24-4, where the minimum lane number is set equal to the lane number of the current lane under consideration.

At step 24-5 it is determined whether the air priorities of the minimum lane and the current lane under consideration are equal. If they are, one of the two lanes must be selected. There are various ways to break this "tie". At step 24-6, this tie is broken based on the first come first served method. Step 24-6 determines whether the minimum lane was ready before the current lane under consideration. If it was not, the program proceeds to step 24-7, where the minimum lane number is set equal to the lane number of the current lane under consideration.

The tie between lanes with equal air priorities could be broken based on other criteria, such as by a fixed sequence or by staged length. The most important thing, is that the possibility of equal air priorities be accommodated.

Following steps 24-2, 24-4, 24-6 or 24-7, the program returns to step 22-14 of the compute the plan program. At step 22-14, the program returns to step 22-5, repeating the loop until all of the plannable lanes have been examined.

At step 22-15 it is determined whether the minimum lane number is zero. If it is zero, then of the lanes remained plannable after step 22-11, and the compute the plan routine is done.

If the minimum lane number is not zero, the minimum lane is inserted as the last entry at step 22-16. At step 22-17, it is determined whether the lane just inserted into the plan has air greater than zero. If its air is greater than zero, the lane is immediately released at step 22-18. At step 22-18, the staging loop is instructed to release and the plan status is set to releasing. At step 22-19, the status of all lanes which have already been inserted into the plan during the current execution of the compute the plan routine are changed from "planned" to "committed".

If the air is not greater than zero, the lane cannot release and the program proceeds to step 22-20, where the lane's wait to release count is set equal to the negative of its air (since the air is negative, this yields a positive wait to release count) and the lane's plan status is set to "planned".

5 The program proceeds to step 22-21 from either step 22-19 or step 22-20. At step 22-21, the last gap to exit count is set equal to the current lane's gap to exit count. At step 22-22, the current lane's lane flag is set to not plannable so that it will not be considered during the subsequent iterative executions of steps 22-2 through 22-21. Steps 22-21 and 22-22 prepare the program to consider the remaining plannable lanes.

10 At step 22-23, the number of plannable lanes are decremented, and the program returns to step 22-2. As is apparent, the compute the plan routine terminates at step 22-24 once there are no more plannable lanes to be considered, either following step 22-2 or 22-15.

 Figures 25A and 25B illustrate the check for blow through routine used in an embodiment of the present invention. This routine is called by step 19-3. When only one
15 lane has packages arriving from the meter belt, there is no need to stage the packages into slugs and coordinate the slug release with other lanes. In such a circumstance, the packages can be released individually to the takeaway conveyor. During blow through, the staging belt receives and releases packages at $V_{SBRELEASE}$ while the meter belt operates between zero and V_{NORM} to make sure that there is at least the minimum gap. Although the speed difference
20 between the meter belt and the staging belt produces some inaccuracy in the package position when the package reaches the staging belt, the minimum gap is ensured by decrementing the tail count for the leading package at V_{NORM} even though the staging belt is traveling faster. This results in the tail count establishing the worst case position (i.e., the furthest upstream position) of the trailing edge of the leading package. Since the meter belt coordinates with
25 the staging belt based on this position, there will always be at least the minimum gap between packages. Excessive gap is not a significant problem since the other lanes are not ready to release when a lane is in blow through mode.

As will be described below, once a lane is in blow through mode, it remains in blow through mode until the status of other lanes knock it out of blow through mode, even though

the blow through lane may not have any packages arriving on it. If there are no packages arriving from a blow through lane, the staging belt and spur do not run continuously.

At step 25-1, it is determined whether the blow through mode is enabled, such as by the operator. If blow through is not enabled, at step 25-2 the blow through lane number is set equal to zero and the routine ends. If blow through is enabled, the program proceeds to step 25-3, where it is determined whether there is a lane currently in blow through mode. If the blow through lane number is not equal to zero, there is a lane in blow through, and the program proceeds to steps 25-4 through 25-11 to determine whether the current blow through mode should be terminated.

At step 25-4, it is determined whether the current blow through lane is disabled. If it is, blow through will be terminated by setting the blow through lane number equal to zero at step 25-12.

If not, the program proceeds to step 25-5, where it is determined whether the current blow through lane is in the maintenance mode. If it is, blow through will be terminated by setting the blow through lane number equal to zero at step 25-12.

If not, the program proceeds to step 25-6, where it is determined whether lockout is on. If any lane is in lockout, lockout is on. If lockout is on, the program proceeds to step 25-7 where it is determined whether the current blow through lane is in lockout. If it is not, blow through will be terminated by setting the blow through lane number equal to zero at step 25-12.

If lockout is not on, or the current blow through lane is not in lockout, the program proceeds to step 25-8, the start of the loop to check the non-blow through lanes. At step 25-9, it is determined whether the non-blow through lane being considered is ok to release. If the lane is, the program proceeds to step 25-10, at which it is determined whether this non-blow through lane is filling. If it is not filling, blow through will be terminated by setting the blow through lane number equal to zero at step 25-12.

If the non-blow through lane being considered is determined at step 25-9 to be not ok to release or is determined at step 25-10 to be filling, the program proceeds to step 25-11 and repeats steps 25-29 and 25-10 for the remaining lanes, unless it is directed to step 25-12 during this loop.

5 From steps 25-3, 25-11 or 25-12, the program proceeds to step 25-13, where it is again determined whether the blow through lane number is equal to zero. If there was a blow through lane at step 25-3, and there is still a blow through lane at step 25-13, it means that none of the conditions for terminating blow through were found to exist in steps 25-4 through 25-11. If there was a blow through lane at step 25-3, but there is not a blow through lane at
10 step 25-13, it means that blow through has been terminated and the program proceeds to steps 25-14 through 25-25 to determine whether to place any lanes into blow through mode. If there was not a blow through lane at step 25-3, the program will proceed through step 25-13 to step 25-14.

 At step 25-14, the blow through count is initialized by setting it equal to zero. Step
15 25-15 is the start of the loop to check all of the lanes. At step 25-16, it is determined whether the lane being considered is in maintenance mode. If it is, the program proceeds to step 25-25 and checks the next lane.

 If it is not, the program proceeds to step 25-17, where it is determined whether the
20 lane being considered is enabled. If it is not, the program proceeds to step 25-25 and checks the next lane.

 If it is, the program proceeds to step 25-18, where it is determined whether lockout is on. If any lane is in lockout, lockout is on. If lockout is on, the program proceeds to step 25-19 where it is determined whether the lane being considered is in lockout. If it is not, the program proceeds to step 25-25 and checks the next lane.

25 If lockout is not on, or the lane being considered is in lockout, the program proceeds to step 25-20, where it is determined whether the lane being considered is in hold for batching mode. If it is, the program proceeds to step 25-25 and checks the next lane.

If it is not, the program proceeds to step 25-21, where it is determined whether the meter belt of lane being considered is empty. If the meter belt is empty, the program proceeds to step 25-25 and checks the next lane.

5 If the meter belt is not empty, the program proceeds to step 25-22, where it is determined whether the staging belt of the lane being considered has a package. If the staging belt does not have a package, the program proceeds to step 25-25 and checks the next lane.

10 If the staging belt has a package, the program proceeds to step 25-23 where the blow through count is incremented. At step 25-24, the temporary lane variable is set equal to the current lane number. The program proceeds to step 25-25 and checks the next lane.

15 After all the lanes have been checked, the program proceeds to step 25-26 at which it is determined whether the blow through count is 1. If it is not one, either no lanes meet the requirements for being placed in blow through mode (blow through count equals zero) or more than one lane met the requirements (blow through count is greater than zero). If only one lane met the requirements, the program proceeds to step 25-27 and the blow through lane number is set equal to the number of the lane that met the requirements.

20 While in blow through mode, if a package reaches its decision point on the meter belt while the staging belt is still releasing a previous package, the meter belt will be allowed to transfer the package to the staging belt. In such a case, the deceleration count is increased and the gap to exit count, the back to exit count and the back to spur count are increased effectively making this next package the last package in a slug.

Figure 26 illustrates the decrement counts routine which check for blow through routine used in an embodiment of the present invention. At step 26-1, the decrement count routine is executed every time there is a tick from the takeaway belt encoder.

25 At step 26-2, the loop to check each lane is started. At step 26-3, it is determined whether this lane is releasing. If the lane is releasing, the program proceeds to step at 26-4,

where it is determined whether the gap to exit count is greater than zero. If it is greater than zero, the gap to exit count is decremented at step 26-5.

5 The program proceeds to step 26-6 from steps 26-4 or 26-5. At step 26-6 it is determined whether the front to exit count is greater than zero. If it is greater than zero, the front to exit count is decremented at step 26-7.

10 The program proceeds to step 26-8 from steps 26-6 or 26-7. At step 26-8 it is determined whether the back to spur count is greater than zero. If it is greater than zero, the gap to exit count is decremented at step 26-9, and the program proceeds to step 26-10. At step 26-10, it is determined whether the back to spur count is equal to zero. If it is equal to zero, the lane's releasing flag is cleared (the plan's lane status is set to complete) at step 26-11.

15 The program proceeds to step 26-12 from steps 26-8, 26-10 or 26-11. At step 26-12 it is determined whether the back to exit count is greater than zero. If it is greater than zero, the back to exit count is decremented at step 26-13, and the program proceeds to step 26-14. At step 26-14, it is determined whether the back to exit count is equal to zero. If it is equal to zero, the lane's plan status is set to complete at step 26-15 and the lane is removed from the plan at step 26-16.

The program proceeds to step 26-22 from steps 26-12, 26-14 or 26-16. At step 26-22 the loop repeats for each lane.

20 If at step 26-3, the lane was determined not to be releasing, the program proceeds to step 26-17, where it is determined whether the lane's plan status is "committed". If it is, the program proceeds to step 26-28, where the wait to release count is decremented. From there, the program proceeds to step 26-19 where it is determined whether the wait to release count is zero. If it is zero, at step 26-20 the program signals the staging belt to release and the lane's plan status is set to "releasing" at step 26-21. The program proceeds to step 26-22 from steps 26-17, 26-19 or 26-21.

25

After all of the lanes have been checked, the decrement counts routine terminates at step 26-23.

Once a slug is released to the takeaway, the system may optionally monitor whether the nose of the slug reaches the takeaway exit photoeye within a predetermined window of time. This can be done by determining whether the takeaway exit photoeye detects a package within a predetermined number of takeaway ticks in front of and behind the front to exit count. If a package does not arrive within this window, a jam is reported and the system downstream is shut down, although packages may continue to be staged but not released to the takeaway. During initial start up, restarts, or if the slug was designated "garbage", this error condition is ignored since the true position of the slug nose is not definitively known.

The conveyor system may be designed for subsequent expansion by the addition of feed conveyor lanes in the future. In such a case, the control system can be set up to accommodate the possible expansion by including the future lanes in the programming. Lanes which are not yet installed can be skipped by the program if they are identified as future lanes. The computer program can easily apply such a test at the appropriate points in the logic.

As described above, step 10-15 calls the determine trailing gap routine. This routine determines the trailing inter-package gap for each package based on its packing algorithm. The practice of various aspects of the present invention can be performed with various packing algorithms, whether currently known, disclosed herein, or developed in the future. For example, in one packing algorithm, inter-package gaps (plus any package attribute based additional inter-package gap such as for wide or tall packages as discussed herein) and inter-slug gaps are fixed.

The trailing gap routine determines the trailing inter-package gaps. The inter-package gap may be the same for every package. It may vary based on various package characteristics or attributes of the package or its neighboring package. For example, if a package is designated a wide package, the trailing inter-package gap may be set greater than the trailing inter-package gaps for non-wide packages. As another example, if a package is designated tall (such as is necessary information to avoid shadowing the trailing package if a

scanning tunnel is used), the trailing inter-package gap may be set greater. Other package attributes may be used in determining the inter-package gap without departing from the teachings of the present invention. Depending on the packing algorithm, the determine trailing gap routine may also determine the inter-slug gap.

5 As discussed above, there are two types of gaps that need to be determined, the inter-package gap and the inter-slug gap. An inter-package gap is necessary to allow the sortation subsystem to direct individual packages to their respective locations. In a zippering type induction subsystem, an inter-package gap is required to allow the packages to be merged together. The length of the inter-package gap is also dictated by the downstream
10 subsystem(s) in which the gap is required, such as a sortation subsystem, scale or scanner. If the inter-package gap required by the sortation subsystem is different than that required by the induction subsystem, there are various ways to adjust the inter-package gaps as the packages travel from one subsystem to another, such as by the use of a gap optimizer. As is well known, excessive inter-package gap reduces overall system throughput.

15 An inter-slug gap is essentially present only in an induction subsystem which is configured to merge slugs together. In a preferred embodiment, one portion of the inter-slug gaps is the trailing inter-package gap of the last package of the slug. The other portion of the inter-slug gaps are required because of variations due to imperfect, inconsistent transfer of packages, particularly when the direction of the packages are changed, such as in saw tooth
20 merges. Inter-slug gaps are not required by the sortation subsystem. Thus, except for the portion attributable to the trailing inter-package gap of the last package of the slug, inter-slug gaps represent unwanted space unless they are removed after they are required.

 The design and operation of gap optimizers are well known to a person of ordinary skill in this art. Gap optimizers attempt to correct gaps within a tolerance range. The average
25 rate that packages flow into a gap optimizer equals the average rate that packages flow out. This means that, for matching input and output speeds, the average inter-package gap on the input side equals the average inter-package gap on the output side. However, to prevent a gap optimizer from inadvertently releasing at a slower rate than its input (resulting in being overfed), in a preferred embodiment, slightly smaller (1 inch to 3 inches smaller) inter-

package gaps were set by the gap optimizer than were set in the merge (after accounting for any speed differences). By way of example, in one embodiment, with the inter-package gap required for a sortation subsystem at 8 inches, the gap optimizer was set to 9 or 10 inches, and the average gap on the takeaway conveyor was set greater than 9 or 10 inches.

5 In the present invention, the average inter-package gap of a slug, from the nose of the slug to the nose of the trailing slug, is dictated by the actual inter-package gaps between the packages of the slug plus the trailing inter-slug gap. The gap optimizer can "redistribute" the trailing inter-slug gap to the actual inter-package gaps as the packages exit the gap optimizer, increasing the actual inter-package gaps while the average inter-package gap remains the same.

10 Figure 27 diagrammatically illustrates this redistribution of the inter-slug gap among the inter-package gaps of the slug. The top row has two slugs of packages, 1-5 and 6-9, illustrating the spacing of the packages and slugs as they arrive at the input of a gap optimizer. The bottom row illustrates the inter-package gaps as the packages exit the gap optimizer. (Package 6 is illustrated on the bottom row to show its position relative to package 5, with packages 7-9 omitted for clarity.) As is shown, the inter-slug gap between packages 5 and 6 has been distributed among the inter-package gaps of packages 1-5. The conveyor system attempts to make these gaps equal to the ideal gap for the downstream subsystem(s), without being less than the minimum gap required.

15 20 Average inter-package gap is the sum of the intended (which includes any package specific inter-package gap variations such as for package attributes) of a group of packages divided by the number of packages in that group (accounting for the last package's trailing inter-package gap).

25 While figure 27 illustrates equal inter-package gaps, it should be noted that any variations in the inter-package gap based on the attributes of a particular package need to be maintained. For example, if extra trailing inter-package gap is included due to its package attributes or the package attributes of a neighboring package, such extra trailing inter-package gap is maintained for that package.

Preferably, downstream of the gap optimizer, the inter-slug gaps cease to exist, with all packages being separated only by the ideal inter-package gap. However, any air which was present between the slugs as they were released to the takeaway conveyor will remain in some form.

5 There is a minimum inter-package gap which a gap optimizer requires in order to adjust the inter-package gaps without decreasing its throughput. In the gap optimizer of a preferred embodiment of the present invention, the working distance within which the gap optimizer can correct the gap is the distance between the transfer section trailing edge of the leading package and the transfer section leading edge of the trailing package. If the necessary
10 correction to the inter-package gap cannot occur within this distance, some gapping precision may be lost.

 There is also a limit as to how much inter-slug gap can be redistributed. Although a gap optimizer can handle packages which are touching, if all the packages within a slug are touching, even with a large trailing inter-slug gap, effectiveness of the gap optimizer may be
15 compromised.

 In accordance an embodiment of the present invention, the inter-package gaps and inter-slug gaps are determined and selected so as to produce a consistent average inter-package gap between packages entering the downstream subsystem(s) which is as close as possible to the ideal inter-package gap required by the downstream subsystem(s). The result
20 is a consistent average inter-package gap delivered to the downstream subsystem(s).

 There are various ways to accomplish the goal of consistent average inter-package gaps. In one way, the minimum inter-slug gap essentially dictates the reduction in the inter-package gaps necessary to achieve an average inter-package gap as close to the ideal inter-package gap as possible. In one packing algorithm, the inter-package gaps are reduced from
25 the ideal inter-package gap by an amount based on the allocation of the minimum inter-slug gap over an estimated number of packages in the slug. This could be a fixed or varying amount. For example, the inter-package gaps could be reduced by 2 inches based on the anticipation of 12 packages, which would offset the minimum inter-slug gap. However, if fewer than 12 packages were staged on the staging belt (such as due to the packages all being

long, or in the case of early planning), the total reduction required to offset the minimum inter-slug gap would not be achieved and there would be more inter-slug gap than was required to be distributed by the gap optimizer to the inter-package gaps to achieve the ideal inter-package gap, thus reducing throughput. If more than 12 packages were staged on the staging belt, the packing algorithm could either stop reducing the inter-package gaps for any package staged after the twelfth package, or it could increase the length of the trailing inter-slug gap for this slug.

As raised by the preceding example, to avoid having not reducing the inter-package gaps enough due to fewer than expected packages, an extra amount can be used. Rather than picking a fixed reduction equal to the reduction dictated by the expected number of packages, the reduction picked could be higher. Once enough packages arrived, the reduction could be stopped. In the above example, rather than the 2 inch reduction as dictated on the expected 12 packages, a 3 inch reduction can be used. Once 8 packages have arrived, all of the reduction will have been applied, and any packages after 9 will not have their inter-package gaps reduced. This avoids having unused or unallocated inter-slug gap due to the arrival of fewer packages.

In accordance with another packing algorithm, the reduction (and therefore the size) of the package's trailing inter-package gap is determined based on its length and the amount of the inter-slug gap remaining to be allocated. In this algorithm, the length of the package on the meter belt is used to predict the total number of packages in the slug. (This algorithm is based on the empirical observation that frequently trailing packages have the same length as the preceding package, making the length of the leading package a good predictor of the length of the trailing package.) As each package arrives, this algorithm determines the number of packages that could still be staged in the remaining available length of the staging belt with inter-package gaps determined based on the ideal inter-package gap being reduced by the "unused" inter-slug reduction being allocated to the available inter-package gaps. Thus, the actual inter-package gap between packages on the staging belt will vary depending on the mix and location of packages while the average inter-package gap resulting from the slug length and the trailing inter-slug gap will be equal to the ideal inter-package gap. To compensate for anomalies, such as shorter slugs in early planning, the inter-package gaps

could be calculated assuming a larger than required inter-slug gap. The appropriate inter-slug gap could then be calculated based on the staging belt population at the time of planning.

It is noted that the inter-slug may be fixed and the same for every lane. Alternatively, the inter-slug gap may vary by lane or even based on slug length. For example, short slugs could be assigned a different inter-slug gap than long slugs.

Another packing algorithm is the dense packing algorithm. Of the various states of the staging belt, filling is the productive time since it is moving packages away from the accumulation conveyor. It is desirable to get the ratio of the time the staging belt is filling to the total cycle time (the fill-cycle time ratio) as high as possible and equalized for each lane. The cycle time is the total time a staging belt spends filling, waiting to be selected, waiting for previously released lanes to clear, and releasing. If the staging belts are equal in length, the time required to fill and release are essentially the same. The time waiting to be selected is also the same from lane to lane. What varies based on lane location is the time spent waiting for upstream lanes to clear this lane's spur. Downstream lanes have to wait to be released until the end of a slug's trailing inter-slug gap is a little upstream of the lane's spur. Thus, the cycle time is higher for a downstream lane, making the fill-cycle time ratio lower for downstream lanes.

One way to equalize the fill-cycle time ratio is to equalize the cycle time for all lanes. This can be done by equalizing the time spent waiting to clear, such as by increasing the length of the spur so that the distances from the staging belt exit to the takeaway exit (or takeaway exit photoeye) for every lane are equal. However, this increases the cost and complexities of the conveyor system and takes up more room. Another way to equalize the fill-cycle time ratio is to increase the staging belt length of the downstream lanes. Longer staging belts also represent increased cost and complexity and require more room.

The dense packing algorithm allows instantaneous control and variation of the fill-cycle ratio. The dense packing algorithm can increase the fill time of a staging belt without having to increase its physical actual length. In the dense packing algorithm, packages are staged on the staging belt with small inter-package gaps and the slug is released at a speed

which is less than the final speed at which the inter-package spacing is required. Because of the speed difference, the inter-package gaps increase as their speed increases.

Figure 28 diagrammatically illustrates the dense packing algorithm from the staging belt to downstream of the gap optimizer. The top row represents packages on the staging belt which have been dense packed. The inter-package gaps are equal to a dense pack gap (including any package specific extra gap due to package attributes), 1 inch for example. Any gap less than the ideal gap can be considered a dense pack gap. The middle row represents the packages after they have been transferred across a speed difference to the final speed (such as transferred to the takeaway conveyor (operating at the final speed) at a slower speed than the takeaway speed) illustrating that the inter-package gaps have increased, for example to 10 inches. As illustrated, the appropriate trailing inter-slug gap is included (which includes the slug's last package's trailing inter-package gap). The bottom row represents the packages downstream of the gap optimizer after the trailing inter-slug gap has been distributed between the packages such that the inter-package gaps are equal to the ideal gap. This illustration shows packages of the same length, which produces identical inter-package gaps on the takeaway conveyor. When packages of different lengths are in a slug, the individual inter-package gaps for the location of the middle row will vary from package to package based on the package length.

The release velocity of the staging belt is given by the formula:

$$V_{SBRELEASE} = V_{FINAL} \times (L_{AVE} + G_{DP}) / (L_{AVE} + G_{FINAL})$$

where V_{FINAL} is the final speed at which the inter-package spacing is required, L_{AVE} is the average package length, G_{DP} is the average dense pack gap length and G_{FINAL} is the average gap between packages at the final speed. If the takeaway conveyor is operating at the final speed, this equation will apply to the inter-package gaps on the takeaway conveyor. It is noted that the takeaway conveyor could be run slower than the final speed which would allow reduce noise and wear. Even if the takeaway conveyor were run at a lower speed than the final speed, the staging belt may still release at a slower speed than the takeaway conveyor's speed. As mentioned, for slugs with packages of different lengths, the actual inter-package gaps will not be identical. However, the gap optimizer will set the inter-package gaps equal

to the ideal inter-package gap (including any package specific extra gap due to package attributes).

In another embodiment of the dense packing algorithm, the staging belt may be broken into logical segments. Each logical segment has its own release velocity which is determined by the above formula as applied to the packages within that segment. The length and number of the segments may vary. For example, a staging belt thirty-six feet long could be broken into four segments nine feet long. The more segments, the more accurate the inter-package gap on the takeaway. The ultimate limit on segments is for each package to represent a logical segment.

The dense packing algorithm can be used to equalize the fill-cycle time ratio between lanes. By setting different dense pack gaps, dense packing for different slug lengths or by varying the total case footage put on the staging belt, variations in the fill-cycle time ratio between lanes can be achieved without changing the physical layout.

The dense packing algorithm can also be used to make the fill-cycle time ratio unequal between lanes for load balancing purposes, setting the fill-cycle time ratios based on the activity level of each lane. If, for example, the operator could place a particular lane in a dense pack mode to accommodate heavy activity from that lane. The amount of dense packing could even be varied from lane to lane, such as by the using different dense pack gaps, by dense packing for different lengths or by varying the total case footage. The advantage of controlling the fill-cycle time ratio to balance loads would be lost if every lane were set to the maximum dense pack capability. When used for load balancing, the dense packing variability could be controlled automatically or manually (using operator controlled inputs).

As is well known to a person of ordinary skill in this art, packages are typically picked in batches. Each picker is provided with a list of packages for the current batch. The packages are picked from respective pick modules, eventually arriving at a respective feed conveyor lane for induction and sortation. Once all of the packages feeding a feed conveyor lane have been picked and placed on the conveyor system, an end of batch tote is sent from the pick module.

When an end of batch tote is detected on a meter belt, the meter belt holds the end of batch tote until the operator signals for it to be released. The accumulation conveyor stops feeding packages to the brake belt, although packages from the next batch will be allowed to accumulate on the accumulator.

5 The operator typically holds all end of batch totes until every active lane has one, and the recirculation lane is empty, signifying that the batch is complete. When the batch is complete, the operator will put additional totes on a feed conveyor lane and release the totes. The computer diverts one tote to each active sortation takeaway line to mark the end of the batch for those removing packages therefrom. Once every sortation takeaway line has
10 received a tote, the extra totes (at least one) proceeds past the end of the sortation subsystem to the recirculation lane. Once the tote returns on the recirculation lane, the operator can start the next batch.

When the entire conveyor, or even just one feed conveyor lane, is started up or restarted, such as when it is powered up at the beginning of a shift or after an emergency stop
15 or a jam, the exact number and location of packages on the staging belt are not reliably known. Although the computer may have data on packages from prior to the previous shut down, such information is not considered reliable. Any time a the contents of a conveyor are not reliably known, i.e., complete accurate information, that conveyor is considered to have "garbage". At start up, including restart, every lane which was shut down or stopped is
20 considered to have garbage.

If the staging belt is considered as having garbage, it is designated as having a slug length equal to the length of the staging belt. An additional trailing inter-slug gap is added to prevent the gap optimizer being overfed. In one embodiment, an additional 100 inches is added.

25 If the meter belt is considered as having garbage, everything between the meter belt photoeye and the end of the meter belt is transferred to the staging belt as a whole unit by setting the first carton length equal to the distance from the product detect photoeye (low detect, high detect, tote lip) to the end of the meter belt. If the photoeye is blocked at start up, then the length of the package being transferred to the staging belt is assumed to be equal to

distance from the be the product photoeye to the end of the meter belt plus the jam length. If during transfer, the product photoeye becomes unblocked before the jam length is exceeded, the meter belt speed will recognize the end of the package and decrease its speed after the package is transferred if necessary order to control the trailing package, but the assumed
5 length of the package on the staging belt remains at the distance from the be the product photoeye to the end of the meter belt plus the jam length. If the photoeye does not become unblocked within the jam length, the system will shut down. If a meter belt transfers garbage to the staging belt, an additional trailing inter-slug gap is added. In a preferred embodiment, the distance from the product photoeyes to the end of the meter belt was 103 inches and the
10 jam length was 73 inches, making the length of the "garbage" transferred to the staging belt 175 inches.

Upon starting the takeaway conveyor, it assumed to be full of garbage, and is run for its entire length, plus an extra gap to avoid overfeeding the gap optimizer. In a preferred embodiment, the takeaway conveyor advanced its "garbage" at a speed less than its normal
15 speed, for example, at 300 feet per minute.

In summary, numerous benefits have been described which result from employing the concepts of the invention. The foregoing description of a preferred embodiment of the invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. Obvious modifications or
20 variations are possible in light of the above teachings. The embodiment was chosen and described in order to best illustrate the principles of the invention and its practical application to thereby enable one of ordinary skill in the art to best utilize the invention in various embodiments and with various modifications as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the claims appended hereto.

1. A conveyor system for conveying a plurality of packages, each of said packages having a head section, a transfer section and a tail section, said conveyor system comprising:
 - 5 a. a meter conveyor
 - b. a staging conveyor
 - c. a computer control operatively connected to said meter conveyor and said staging conveyor so as to control the speeds thereof, said computer control being programmed to vary the speed of said metering belt between zero and a
10 first speed and to vary the speed of said staging belt between zero and a second speed, said computer control also programmed to coordinate the respective speeds of said meter conveyor and said staging conveyor such that when said packages are transferred from said meter conveyor to said staging conveyor the respective speeds of said meter conveyor and said staging
15 conveyor are substantially equal during the time that said transfer sections of packages are transferred.