Propuesta de logica transportadores:

1. Se crea un solo bloque AOI, este integrara la logica que permite funcionar de la siguiente manera
   1. Transportador de entrada-INFEED
   2. Trasportador intermedio-TRANSFER
   3. Transportador de salida-OUTFEED
   4. Este funcionamiento depende de un parametro INT, 1-2-3 respectivamente.
2. Este debe integrar configuración para su uso, que se detalla a continuación:
   1. Configurar cuantos sensores se encuentran en un solo transportador
      1. Un solo sensor en transportador zona de salida. La ventaja es menor costo, la desventaja es menor control al momento de inicializar el sistema.
      2. Dos sensor, uno en la mitad del transportador y otro en la salida, deventaja, mayor costo, ventaja mayor control del objeto a desplazar, permite conocer de inmediato el status al inilizar el sistema.
      3. Se recomienda tener los dos combinados.
3. Debe tener configuracion para definir el modo de trasporte de los objetos de forma combinada.
   1. Sincrona : mueve los objetos todos a la vez mientras haya un transportador libre en sentido de aguas abajo, no requiere que el siguiente transportador se encuentre vacio, comienza a mover el objeto cuando el siguiente esta entregando.
   2. Sequencial, mueve de espacio a espacio, quiere decir que debe estar liberado el sistema siguiente, vacio o sin producto, para que se mueva.
4. Señal externa para bloquear/liberar el transportador, señal booleana que permite el bloqueo del transportador, util para el transportador que se encuentra en una posicion donde trabaja el robot u otro componente
5. Entrada de mapa y salida de mapa, la informacion será entregada por una salida mapeada, esto con tal de poder entregar los estatus de los transportador hacia el HMI, ademas esta misma puede ser compartida entre bloques AOI para que todos conozcan el status del transportador anterior y posterior, ayudando a la sincronizacion de la logica.
6. Todos estos puntos anteriores, ayuda a estandarizar el sistema, ademas permite la escalabilidad.

Lo status que deber tener los AOI para lograr realizar el trabajo combinado, seran lo siguientes, OUT\_Mapping.

1. AutoMode- OUT\_Mapping.0
2. ManualMode- OUT\_Mapping.1
3. ServiceMode OUT\_Mapping.2
4. ReadyToReceive OUT\_Mapping.3
5. Ocupied OUT\_Mapping.4
6. Delivering OUT\_Mapping.5
7. Receiving OUT\_Mapping.6
8. Blocked/Free OUT\_Mapping.7
9. Starved OUT\_Mapping.8
   1. parecido a ready to receive pero sirve para reportes, ya que indica que esta todo listo pero no hay suministro aguas arriba
10. TimeoutReceiving OUT\_Mapping.9
    1. nunca llego el producto
11. TimeOutDelivering OUT\_Mapping.10
    1. nunca llego la confirmacion de llegada del producto

Salidas Booleanas

* Out\_StartMotor BOOL- OUT\_Mapping.11

Los comandos que el AOI recibira seran:

* IN\_ConfigSensor-INT
* IN\_TransferMode-INT
* IN\_Mapping-INT
* IN\_ForceOccupied-BOOL
* IN\_ ReadyToReceive-BOOL
* ES-emergency Stop-BOOL
* IN\_SetWaitTimeReceived
* IN\_SetWaitTimeDelivering
* IN\_SetDelayReceiving
* IN\_SetDelayDelivering

Un programa tipico deberia verse como sigue:

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.



Lo anterior muestra 4 transportadores, donde arriba contiene la configuracion para todos ellos.

El conveyor 2 es el unico configurado con bloqueo externo (IN\_BlockConveyors), marcado en azul, para liberarlo es necesario un pulso que llega desde una fuente externa, como se ve abajo.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Arriba se aprecia mas claramente como quedan los AOI configurados, las salidas pueden ser usadas o simplemente sirven para visualizar de una forma mas rapida el status de cada transportador desde Studio 5000