MultiBox Palletizer Project

Ange Derick Nzangue
Zijian Jiang
Elshan Moradkhani

University of Bologna, DEI

Introduction

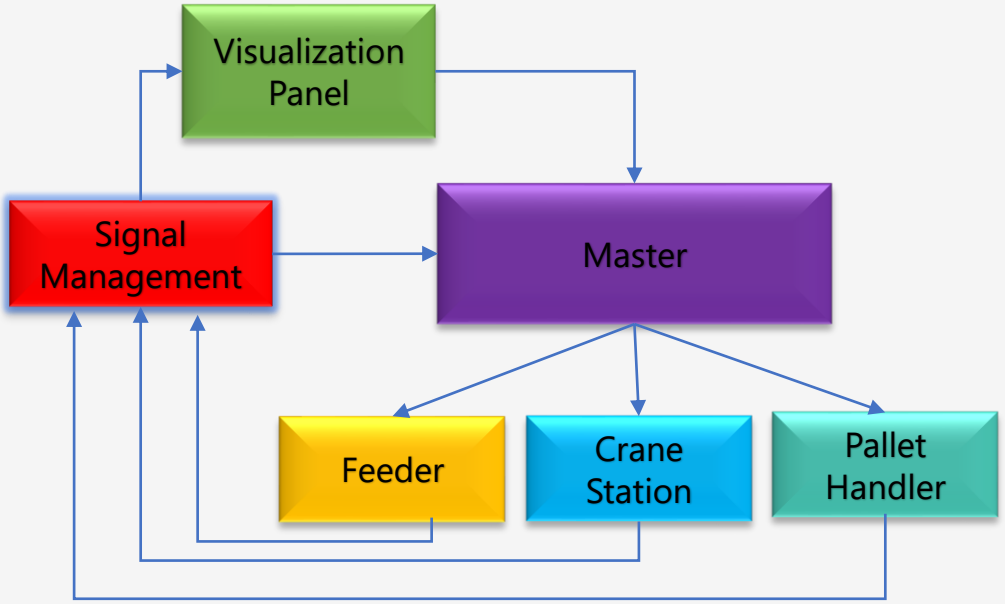
Master:
It oversees the overall functioning of the machine by exchanging signals with the feeder, crane, and pallet handler.

Feeder:
It creates the first and second layers of pallets by appropriately adjusting the orientation of the boxes. It also inserts the products for collection.

Crane Station:
It transports the completed layer of boxes from the feeder to the pallet. During this process, it adds a cardboard under each box in the layer to ensure stability.

Pallet Handler:
It retrieves a new pallet from the warehouse and positions it beneath the crane, awaiting the loading of two layers of boxes. Once loaded, it allows the pallet to exit the machine.

Visualization Panel / HMI:
It is used to visualize the acquired signal and also to commanding the master



Signal and Fault Management:
managing communication signals and addressing system abnormalities to ensure proper functioning and prevent failures.

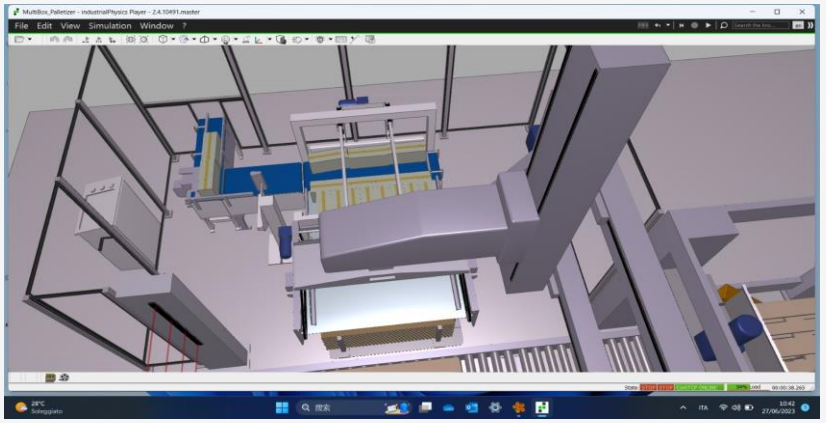
Introduction

The Machine is composed of three units:

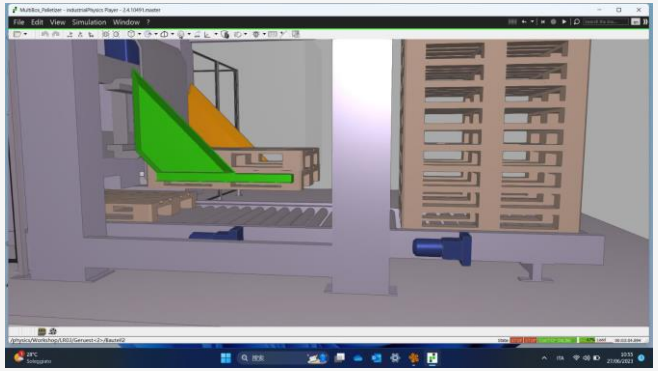
- Feeder
- Crane Station
- Pallet Handler

❑ The main objective of this Machine is to create 2-layer of boxes with different orientations and put them on a pallet.

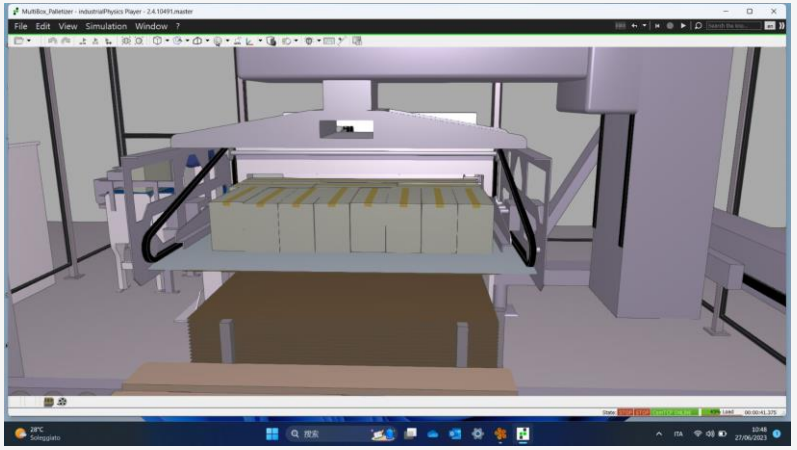
❑ Cartboard must be placed between every two layers.



Feeder

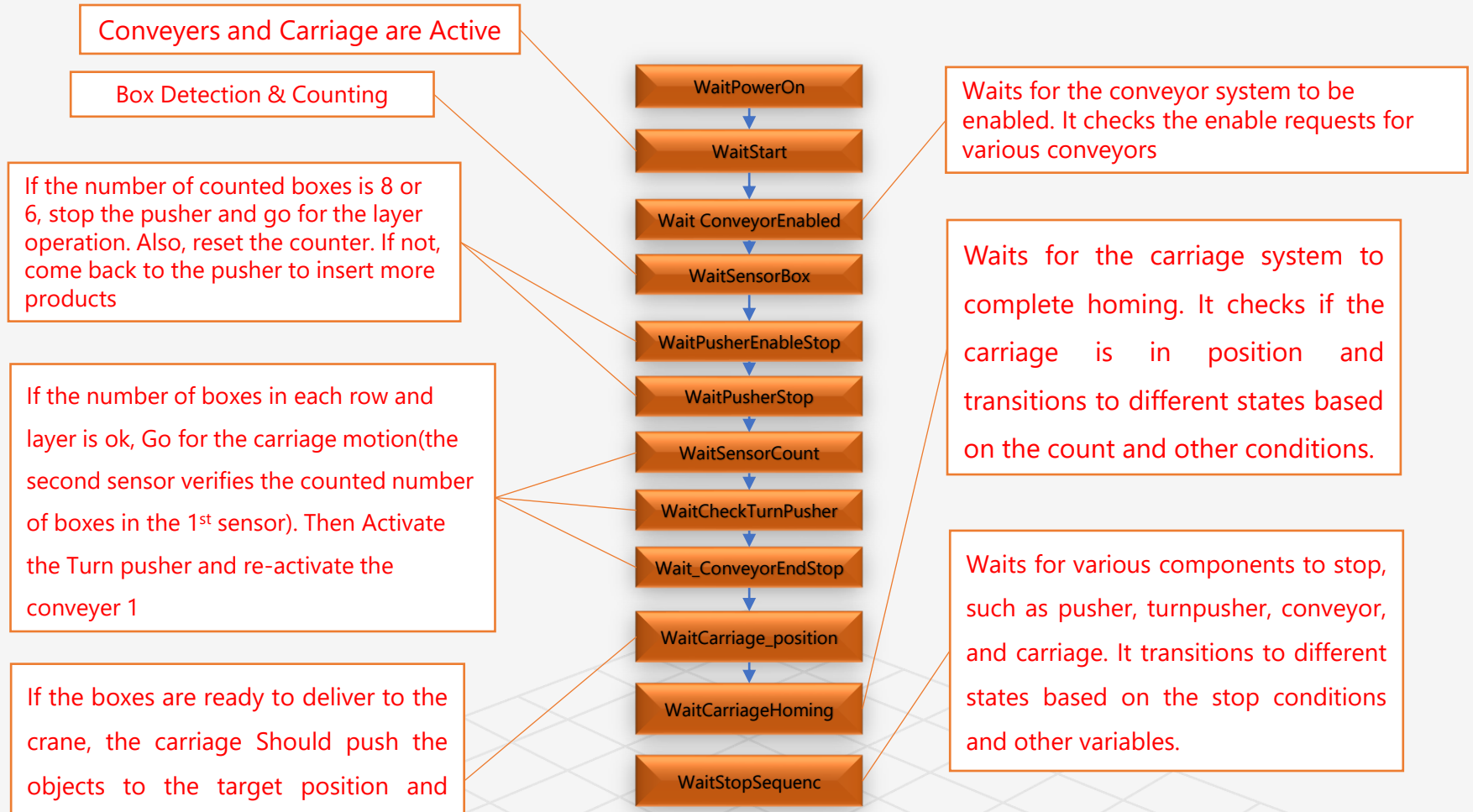


Pallet Handler



Crane Station

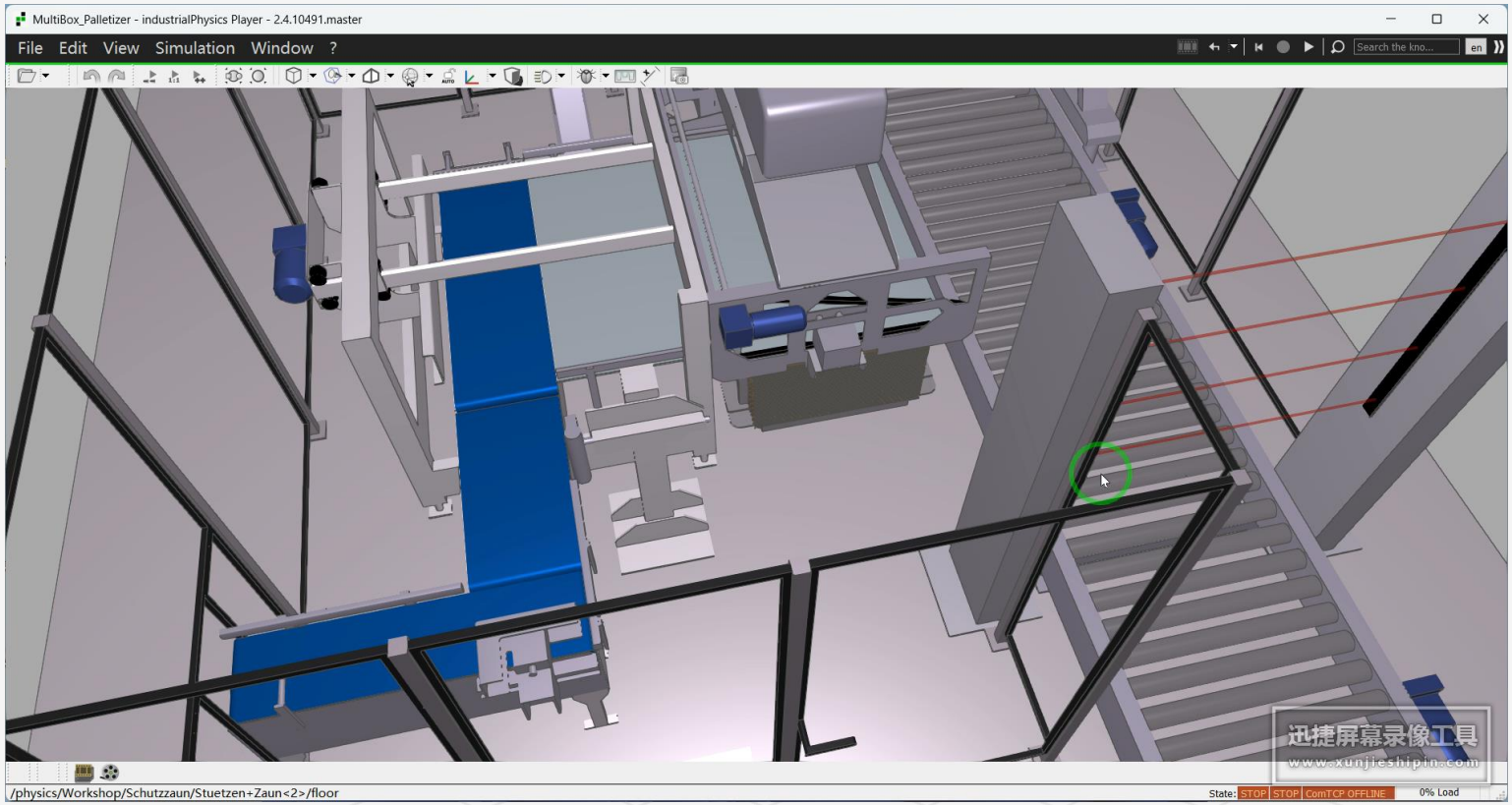
Feeder


Conveyers and Carriage are Active
Box Detection & Counting
If the number of counted boxes is 8 or 6, stop the pusher and go for the layer operation. Also, reset the counter. If not, come back to the pusher to insert more products
If the number of boxes in each row and layer is ok, Go for the carriage motion(the second sensor verifies the counted number of boxes in the 1st sensor). Then Activate the Turn pusher and re-activate the conveyer 1
If the boxes are ready to deliver to the crane, the carriage Should push the objects to the target position and compare it with the Actual position based on the index value.
WaitPowerOn
WaitStart
Wait ConveyorEnabled
WaitSensorBox
WaitPusherEnableStop
WaitPusherStop
WaitSensorCount
WaitCheckTurnPusher
Wait_ConveyorEndStop
WaitCarriage_position
WaitCarriageHoming
WaitStopSequenc
Waits for the conveyor system to be enabled. It checks the enable requests for various conveyors
Waits for the carriage system to complete homing. It checks if the carriage is in position and transitions to different states based on the count and other conditions.
Waits for various components to stop, such as pusher, turnpusher, conveyor, and carriage. It transitions to different states based on the stop conditions and other variables.

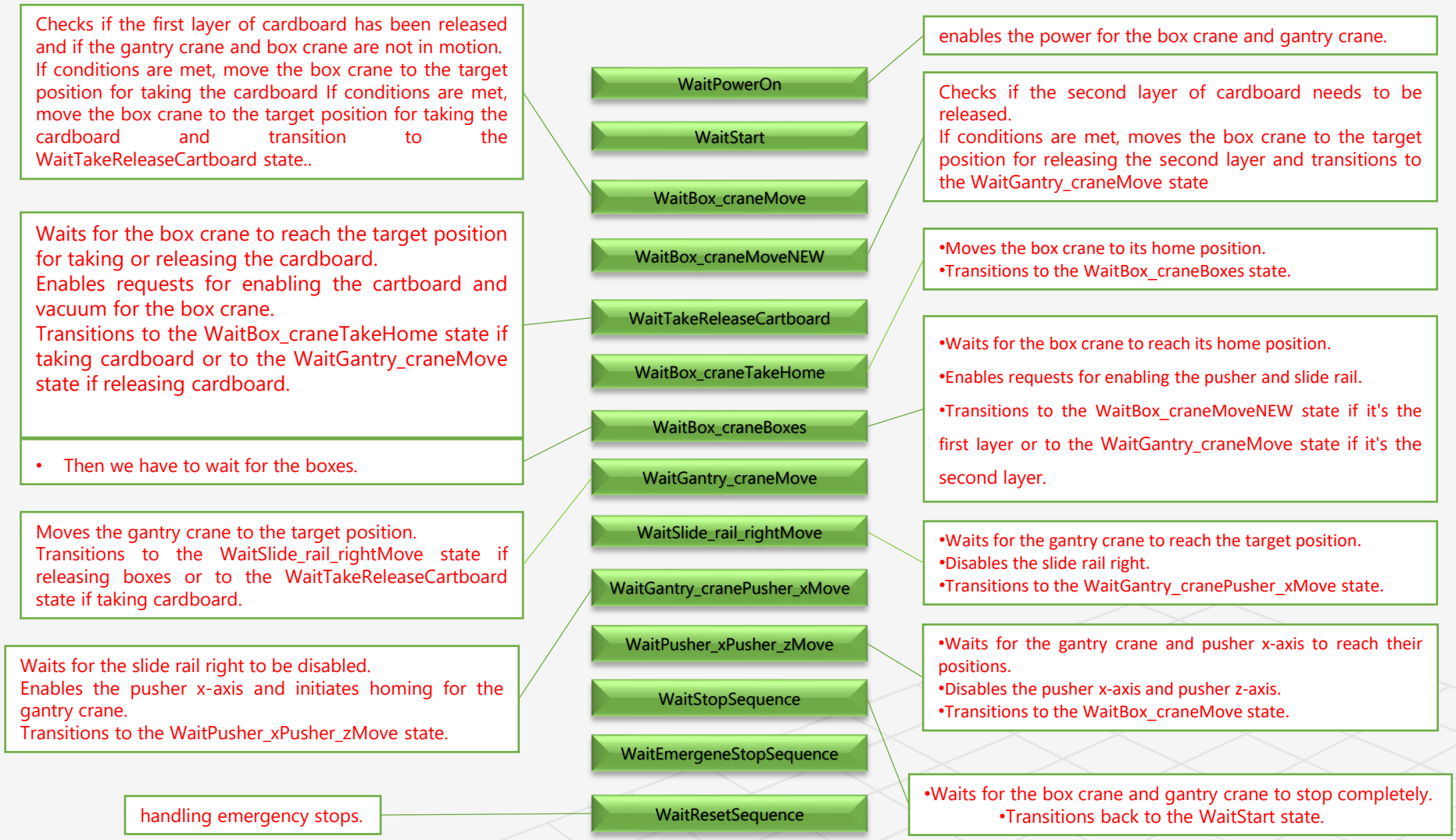
Feeder

- **WaitPowerON:** Waits for power to turn on.
- **WaitStart:** Transitions to WaitConveyorEnabled if *START_FEEDER* is true; else remains in WaitStart.
- **WaitConveyorEnabled:** Transitions to WaitSensorBox on specific conveyor enable conditions.
- **WaitSensorBox:** Transitions to WaitpusherenableStop if LASER_SENSOR_IR is true.
- **WaitpusherenableStop:** Transitions to WaitPusherStop based on WaitCarriage and FIRST_LAYER_OPERATION variables.
- **WaitPusherStop:** Transitions to WaitSensorCount if pusher_enable_request is false.
- **WaitSensorCount:** Transitions to WaitCarriage_position or WaitConveyorEnabled based on NUM_BOX_COUNTED and FIRST_LAYER_OPERATION.
- **WaitCheckTurnPusher:** Transitions to WaitConveyor_endStop based on CountPusherState conditions.
- **WaitConveyor_endStop:** Transitions to WaitCarriage_position if TimeOut is true and specific conditions are met.
- **WaitCarriage_position:** Transitions to WaitCarriage_Homing if carriage_2.InPosition is true.
- **WaitCarriage_Homing**: Transitions to WaitConveyorEnabled based on carriage_2.InPosition, ACTUAL_BOXES, oddsequence, and START_AGAIN_FEEDER variables.
- **WaitStopSequence:** Transitions to WaitPowerON or WaitStart based on pusher_disable_request, conveyor_turn1_disable_request, conveyor_end_disable_request, and carriage_2.InVelocity conditions.

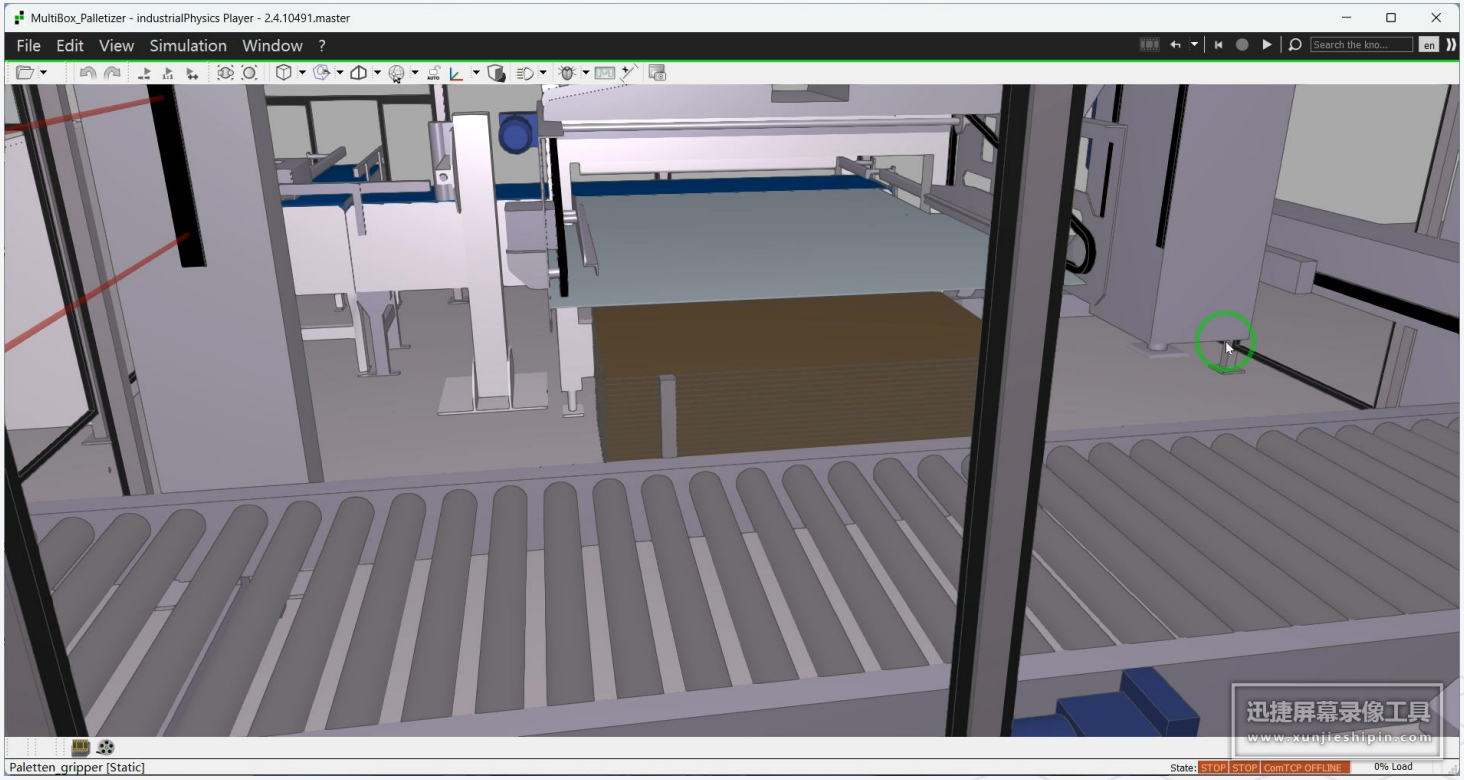
Feeder Simulation (Video)



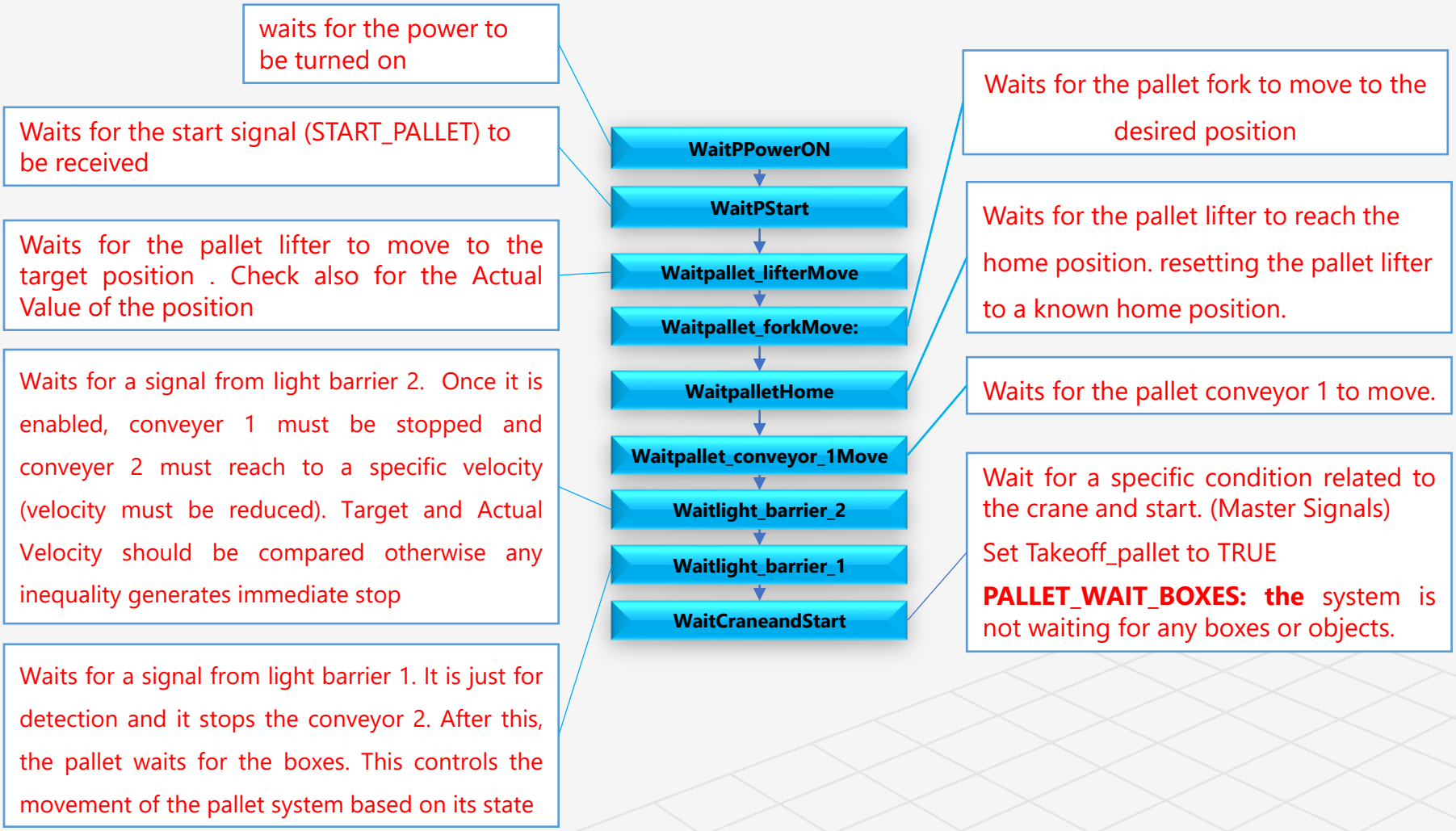
Crane


Checks if the first layer of cardboard has been released and if the gantry crane and box crane are not in motion.
If conditions are met, move the box crane to the target position for taking the cardboard If conditions are met, move the box crane to the target position for taking the cardboard and transition to the WaitTakeReleaseCartboard state..
Waits for the box crane to reach the target position for taking or releasing the cardboard.
Enables requests for enabling the cartboard and vacuum for the box crane.
Transitions to the WaitBox_craneTakeHome state if taking cardboard or to the WaitGantry_craneMove state if releasing cardboard.
• Then we have to wait for the boxes.
Moves the gantry crane to the target position.
Transitions to the WaitSlide_rail_rightMove state if releasing boxes or to the WaitTakeReleaseCartboard state if taking cardboard.
Waits for the slide rail right to be disabled.
Enables the pusher x-axis and initiates homing for the gantry crane.
Transitions to the WaitPusher_xPusher_zMove state.
handling emergency stops.
WaitPowerOn
WaitStart
WaitBox_craneMove
WaitBox_craneMoveNEW
WaitTakeReleaseCartboard
WaitBox_craneTakeHome
WaitBox_craneBoxes
WaitGantry_craneMove
WaitSlide_rail_rightMove
WaitGantry_cranePusher_xMove
WaitPusher_xPusher_zMove
WaitStopSequence
WaitEmergeneStopSequence
WaitResetSequence
enables the power for the box crane and gantry crane.
Checks if the second layer of cardboard needs to be released.
If conditions are met, moves the box crane to the target position for releasing the second layer and transitions to the WaitGantry_craneMove state
•Moves the box crane to its home position.
•Transitions to the WaitBox_craneBoxes state.
•Waits for the box crane to reach its home position.
•Enables requests for enabling the pusher and slide rail.
•Transitions to the WaitBox_craneMoveNEW state if it's the first layer or to the WaitGantry_craneMove state if it's the second layer.
•Waits for the gantry crane to reach the target position.
•Disables the slide rail right.
•Transitions to the WaitGantry_cranePusher_xMove state.
•Waits for the gantry crane and pusher x-axis to reach their positions.
•Disables the pusher x-axis and pusher z-axis.
•Transitions to the WaitBox_craneMove state.
•Waits for the box crane and gantry crane to stop completely.
•Transitions back to the WaitStart state.

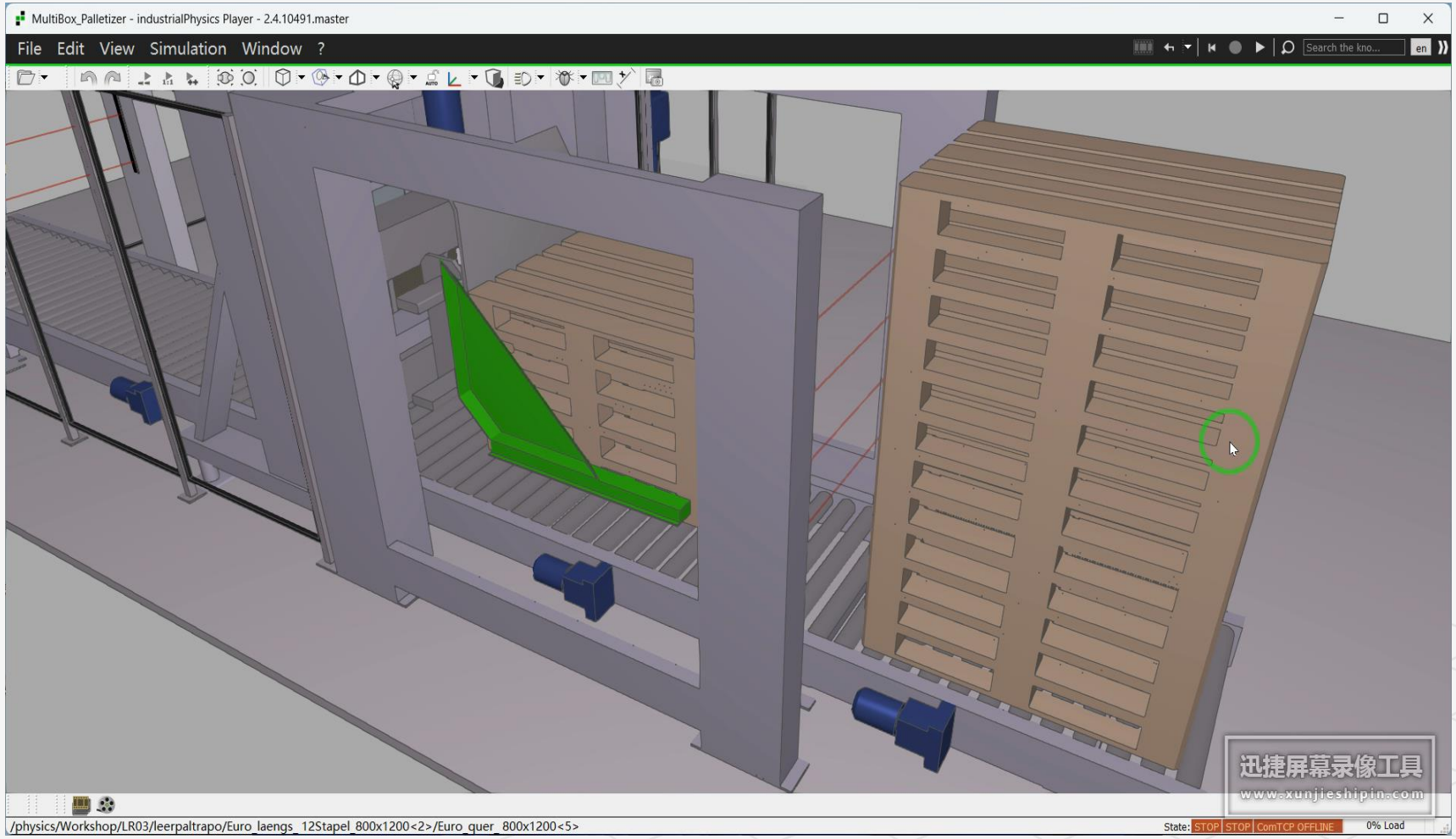
Crane Simulation (Video)



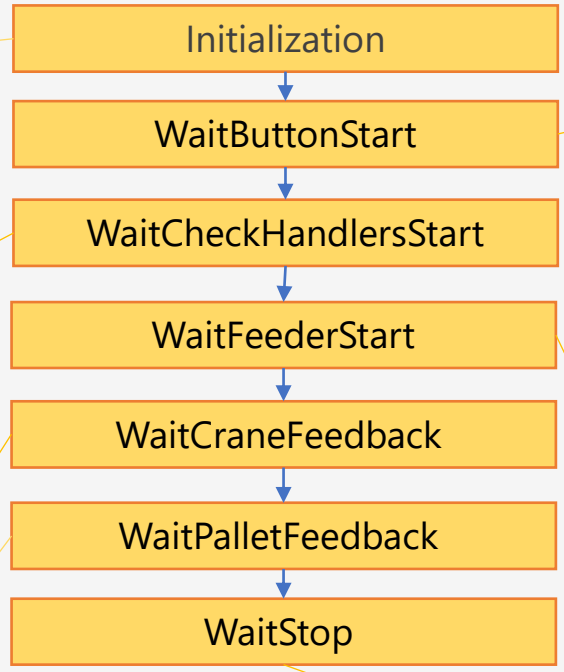
Pallet Handler FSM and Simulation



Pallet Handler FSM and Simulation



Main (Master) FSM



initializing variables and setting initial values. It resets flags related to various buttons and initializes blink flags

waits for the **BUTTON_START** to be activated.

If the BUTTON_START is pressed, it **sets various flags** and variables related to the system operation.

It also sets the blink flags for start and stop buttons.

If all handlers (**FEEDER, CRANE,** and **PALLET**) have started.

If all handlers have started, it transitions to the WaitFeederFeedback state.

If any of the handlers have not started, it continues to stay in this state.

This state waits for feedback from the FEEDER.
If the **FEEDER_WAIT_TO_DELIVER_BOXES** flag is **true**, it indicates that the feeder is ready to deliver boxes.
It sets the **Onelayeroperation** variable to **1** and transitions to the **WaitCraneFeedback state**.
If the **FEEDER_BOXES_DELIVERED** flag is **true** and **START_AGAIN_FEEDER** is **false**, it indicates that the feeder has delivered boxes and it's not required to start again.
It sets the **START_AGAIN_FEEDER** to **true**, resets the **FEEDER_BOXES_DELIVERED** flag, sets **Onelayeroperation** to 2.

If the **CRANE_WAIT_BOXES** flag is **true** and **Onelayeroperation** is **1** or **2**, it indicates that the crane is waiting for boxes.

If **Onelayeroperation** is **2**, it means the boxes delivered in the previous cycle and sets **CRANE_WAIT_BOXES** to **false**.

If the **PALLET_WAIT_BOXES** flag is **true**, it indicates that the pallet is waiting for boxes.

If **Onelayeroperation** is **1**, it means the boxes are waiting at the feeder. It transitions back to the **WaitFeederFeedback** state.

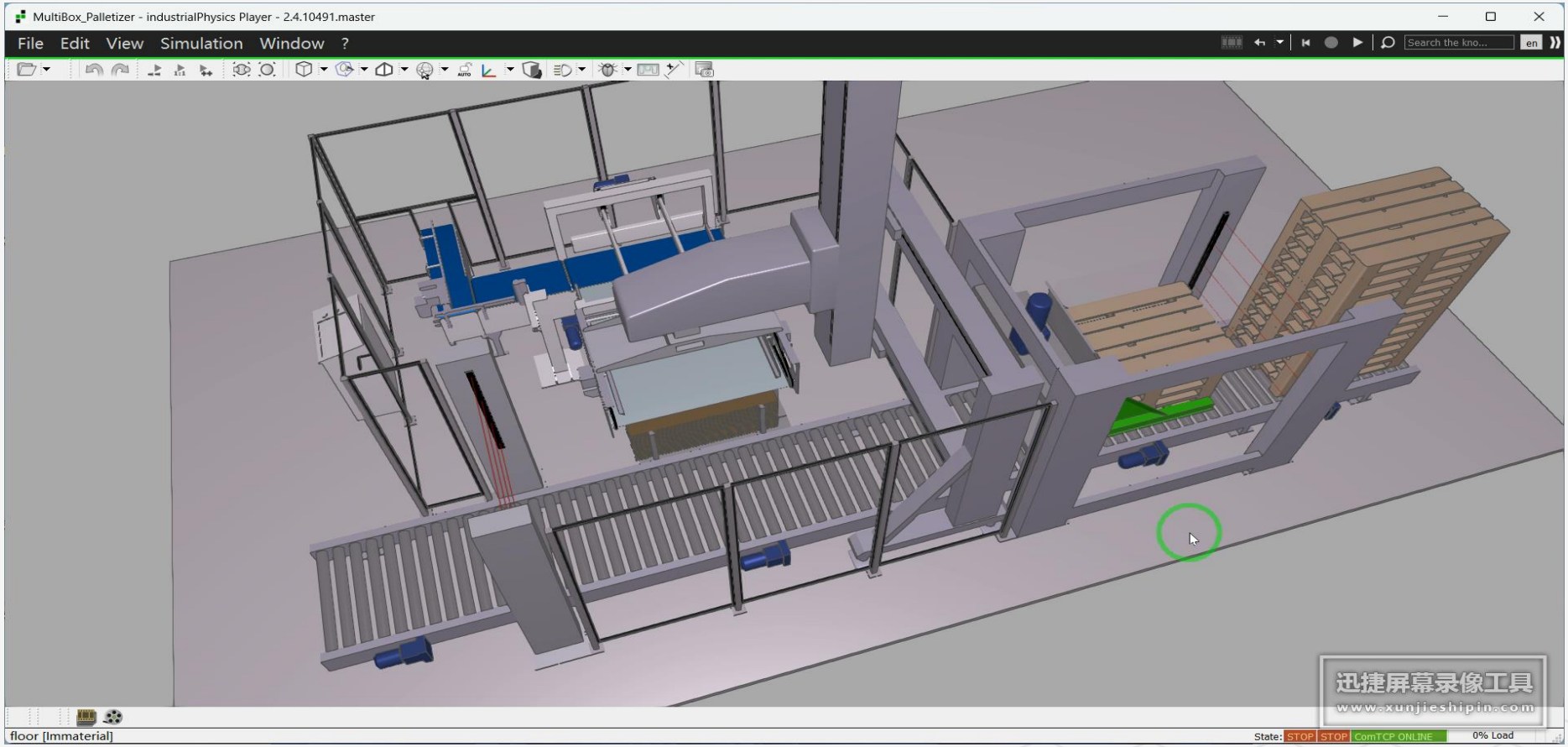
If **Onelayeroperation** is **2** and **CRANE_BOXES_DELIVERED** is **true**, it means the boxes are delivered by the crane, and it transitions back to the **WaitFeederFeedback** state after performing necessary operations and resetting flags.

WaitOnphaseStop, WaitImmediateStop, WaitEmergenceStop, WaitStopSequence, WaitReinitializationSequence: These states are defined in the code but are not implemented in
They could be used for specific conditions or actions but haven't been filled with specific logic in the provided code.

Main (Master) FSM

- **S1:** Power On
- **S2:** Check Everything like Materials: Pallet, Car, Box …
- **S3:** Move the Pallet to the Crane to be prepared.
- **S4:** Start Motion of the Feeder
- **S4:** Move Crane to pick a Cart
- **S5:** Start Motion of the Feeder
- **S6:** Feeder Signals the Master,
  The boxes are Ready to be delivered to the crane then we have to Restart the Feeder Again
- **S7:** Crane Signal the Master
  Start again the Pallet form its Station
  Keep Feeder Working
- **S8:** Signal the from Pallet
  Start again the Crane

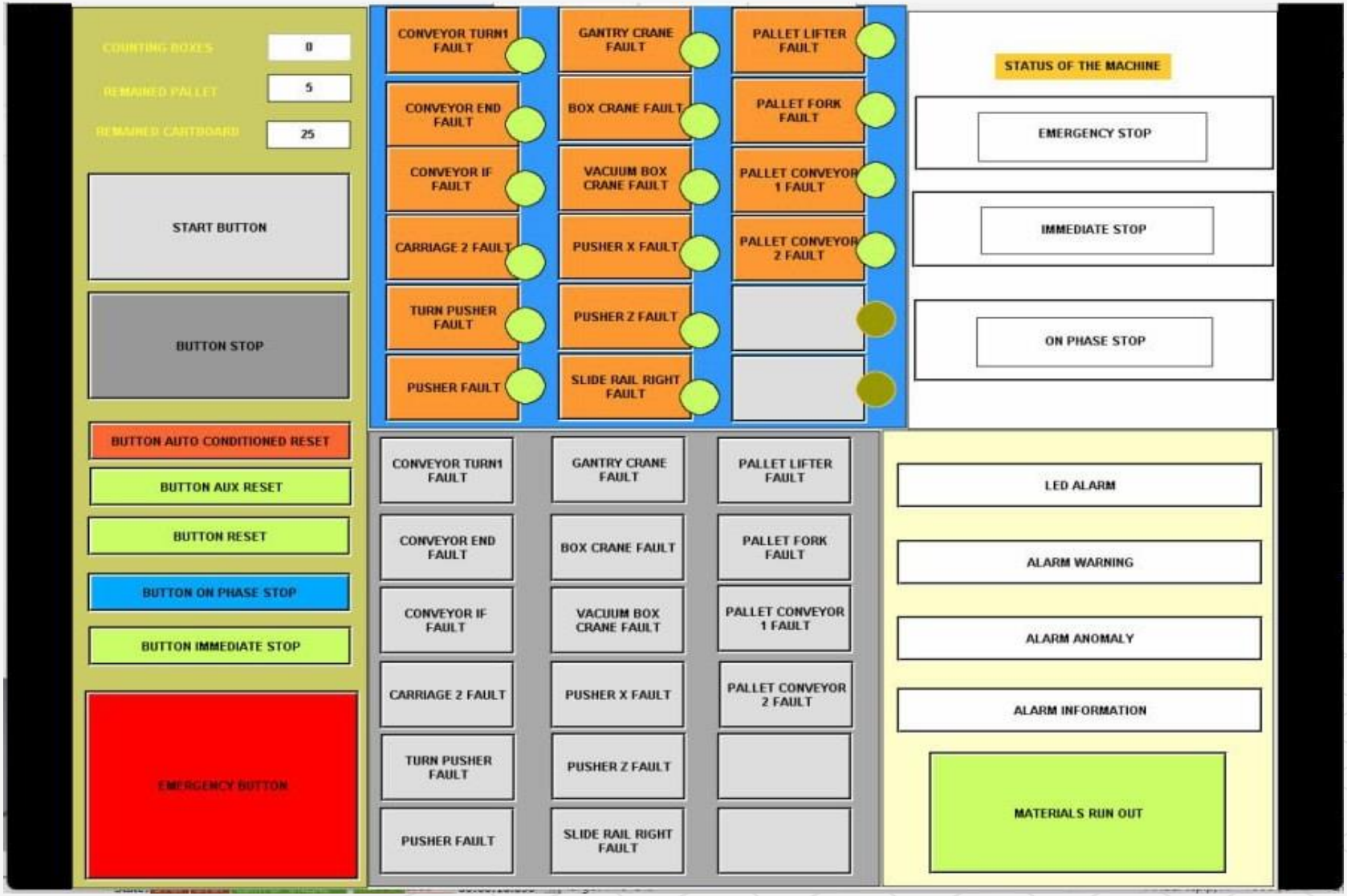
Machine Simulation



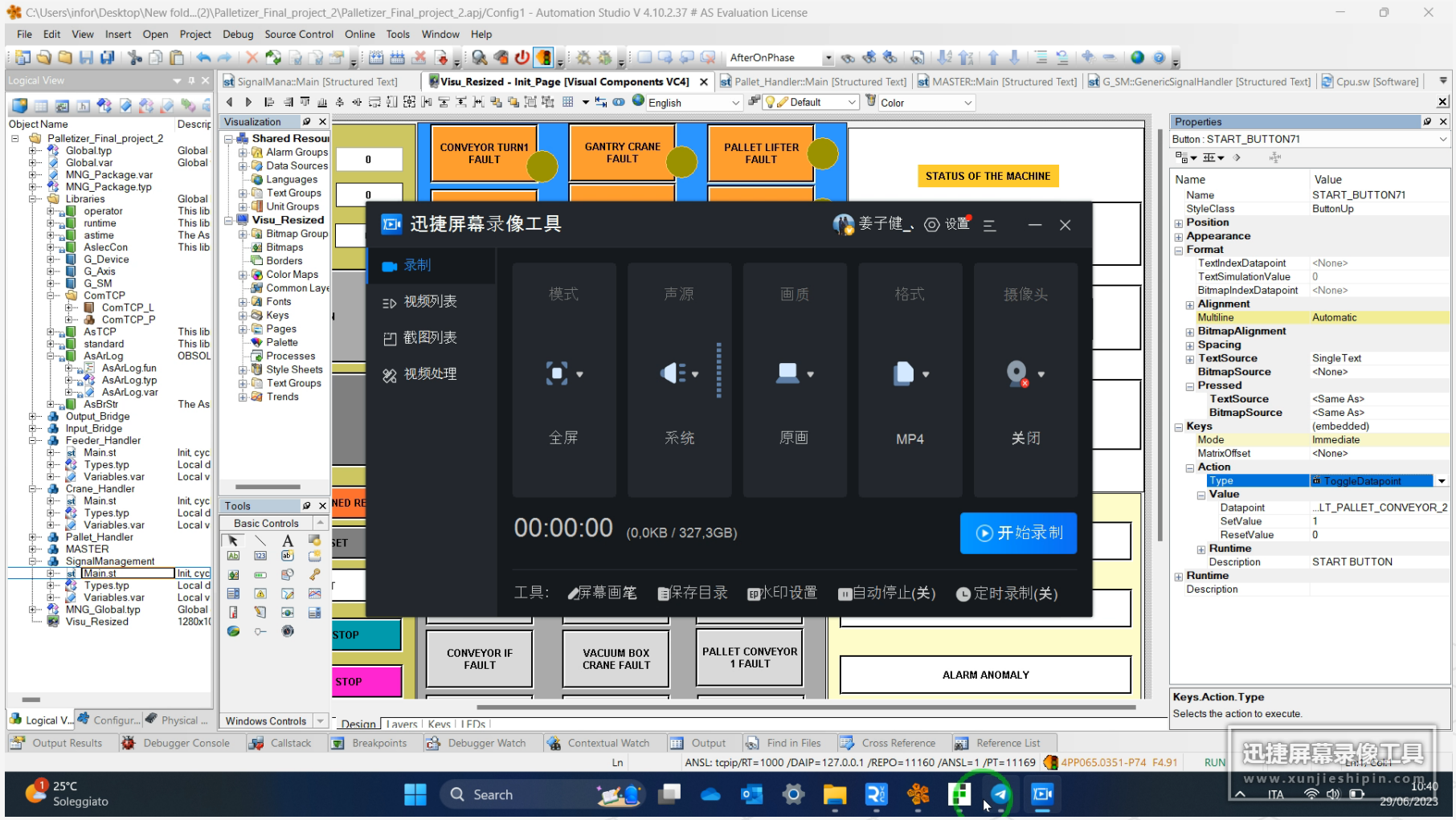
Signal and Error Management

- Emergency Stop:
  - The entire machine has come to a halt.
  - An Auxiliary Reset is required.
  - Following the Auxiliary Reset, the machine will undergo re-initialization.
  - The Emergency Stop has got a greater priority than the On Phase Stop.
  - The Emergency Stop has got also a higher priority than the Immediate Stop.
- Immediate Stop:
  - Each time it happens, the entire machine comes to a halt. The current state of the machine is preserved.
  - An Auxiliary Reset is required.
  - Following the Auxiliary Reset, the machine will resume operation from the exact state at which it was stopped.
  - The Immediate Stop has got a greater priority than the On Phase Stop.
- On Phase Stop:
  - Whenever it occurs, the entire machine will reach a secure state and subsequently come to a stop. The current status of the machine is saved.
  - A reset is required.
  - On phase Stop: material over.
  - Following the Reset, the machine will restart from the safe state it reached prior.
- Warning Lights:
  - They light to signal the occurrence of a warning, such as low material levels.

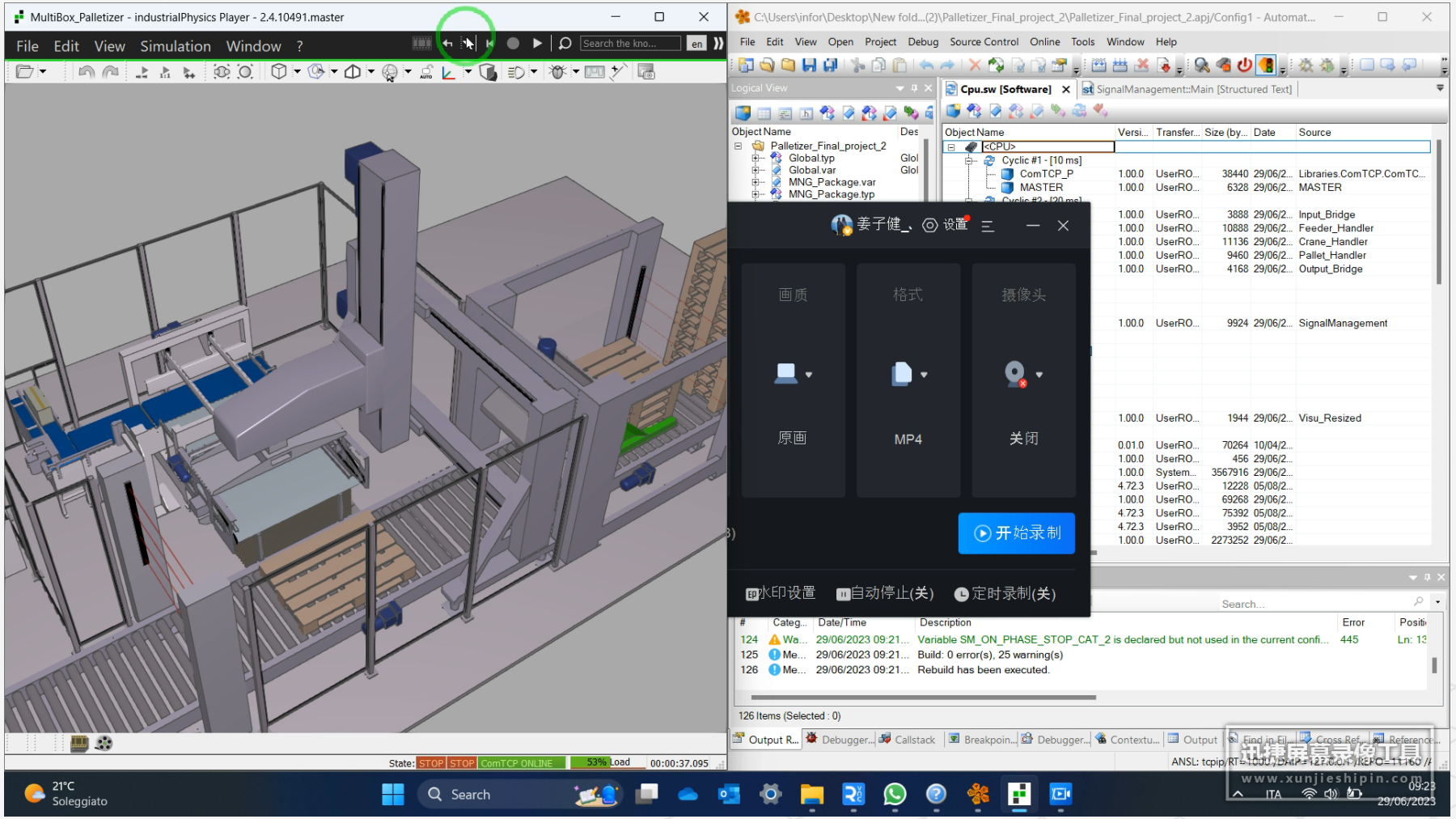
Visualization / HMI



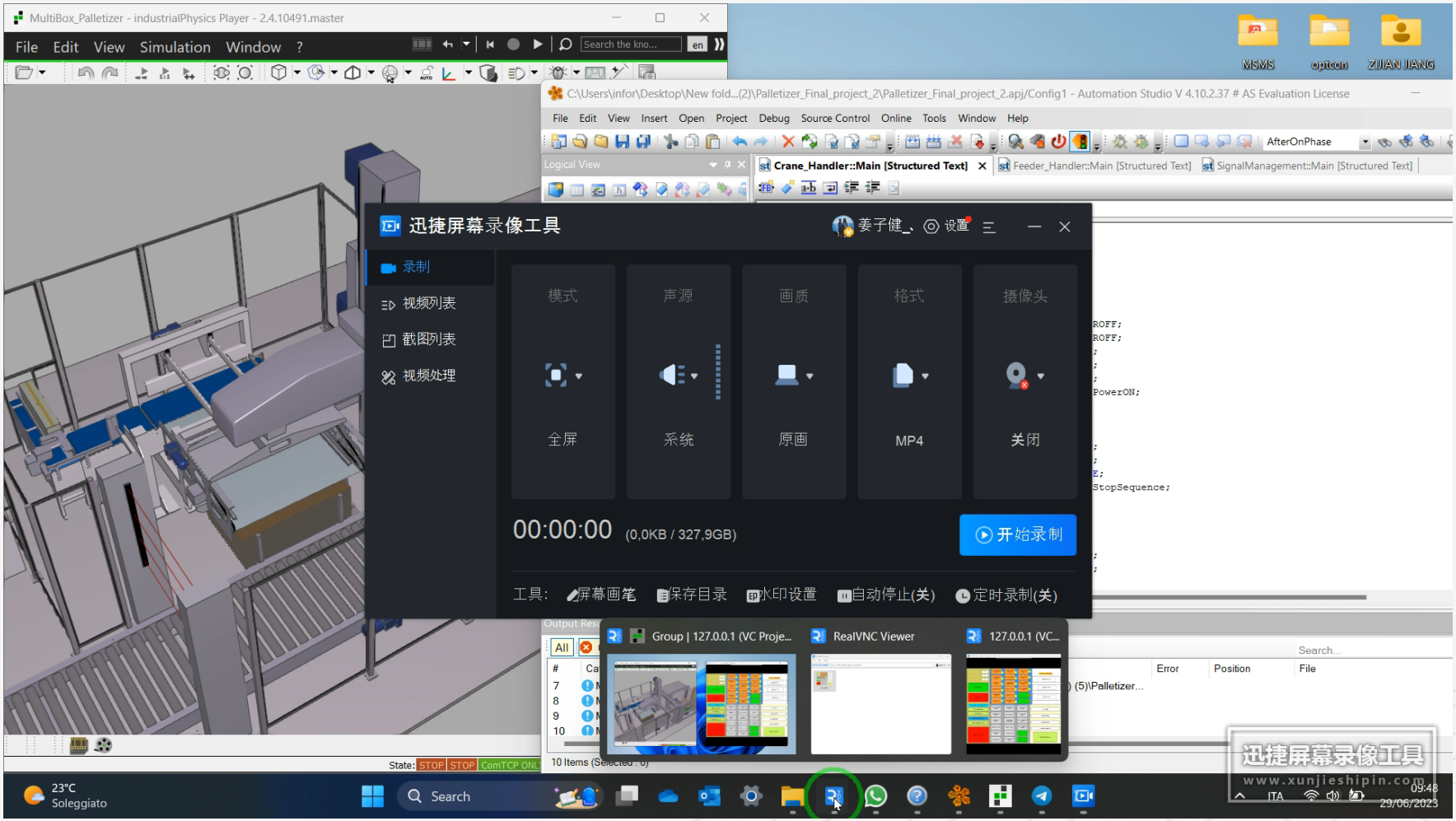
Signal and Error Management Simulation (Component Error)



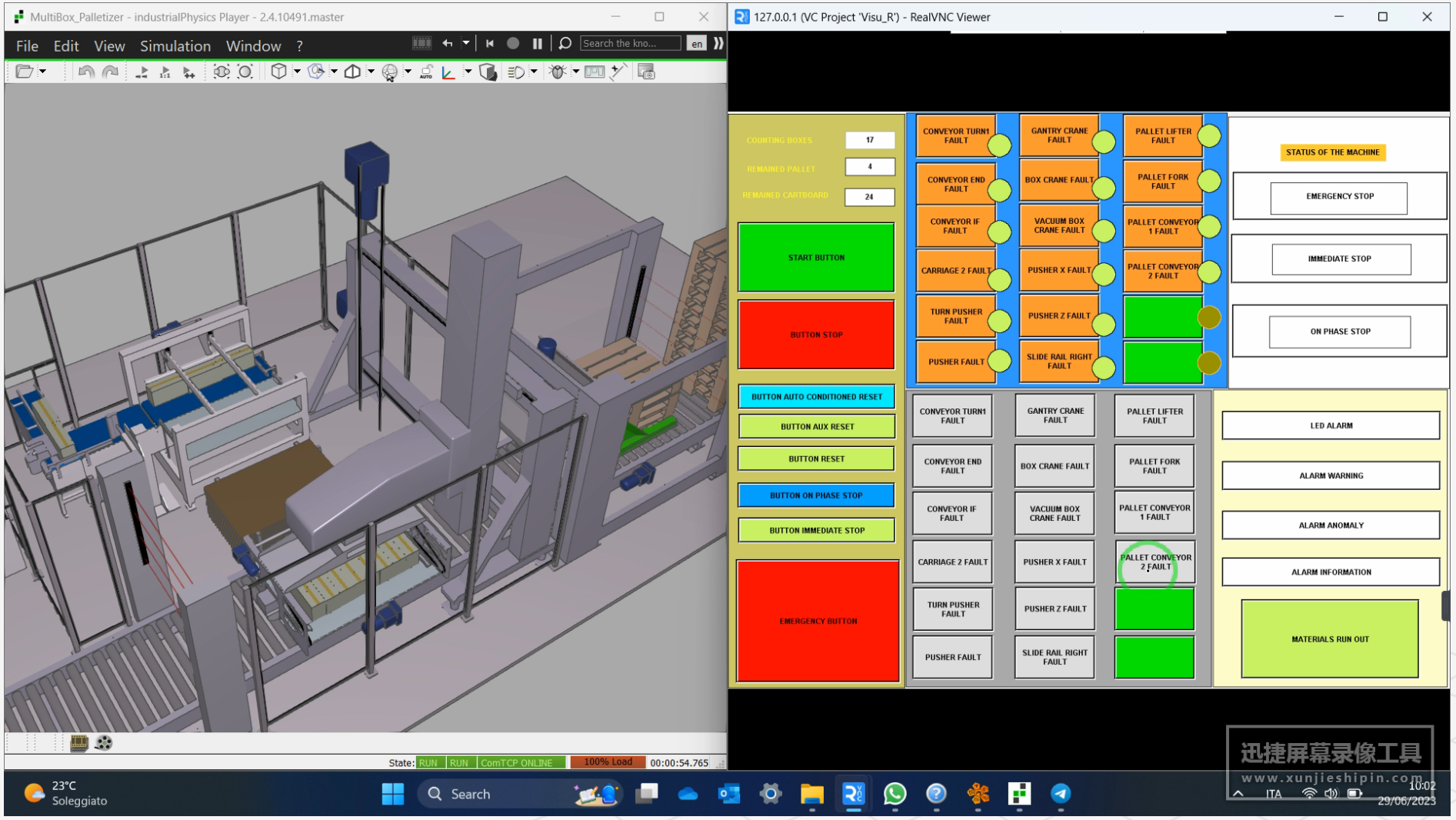
Signal and Error Management Simulation (Out of Material Alarm)



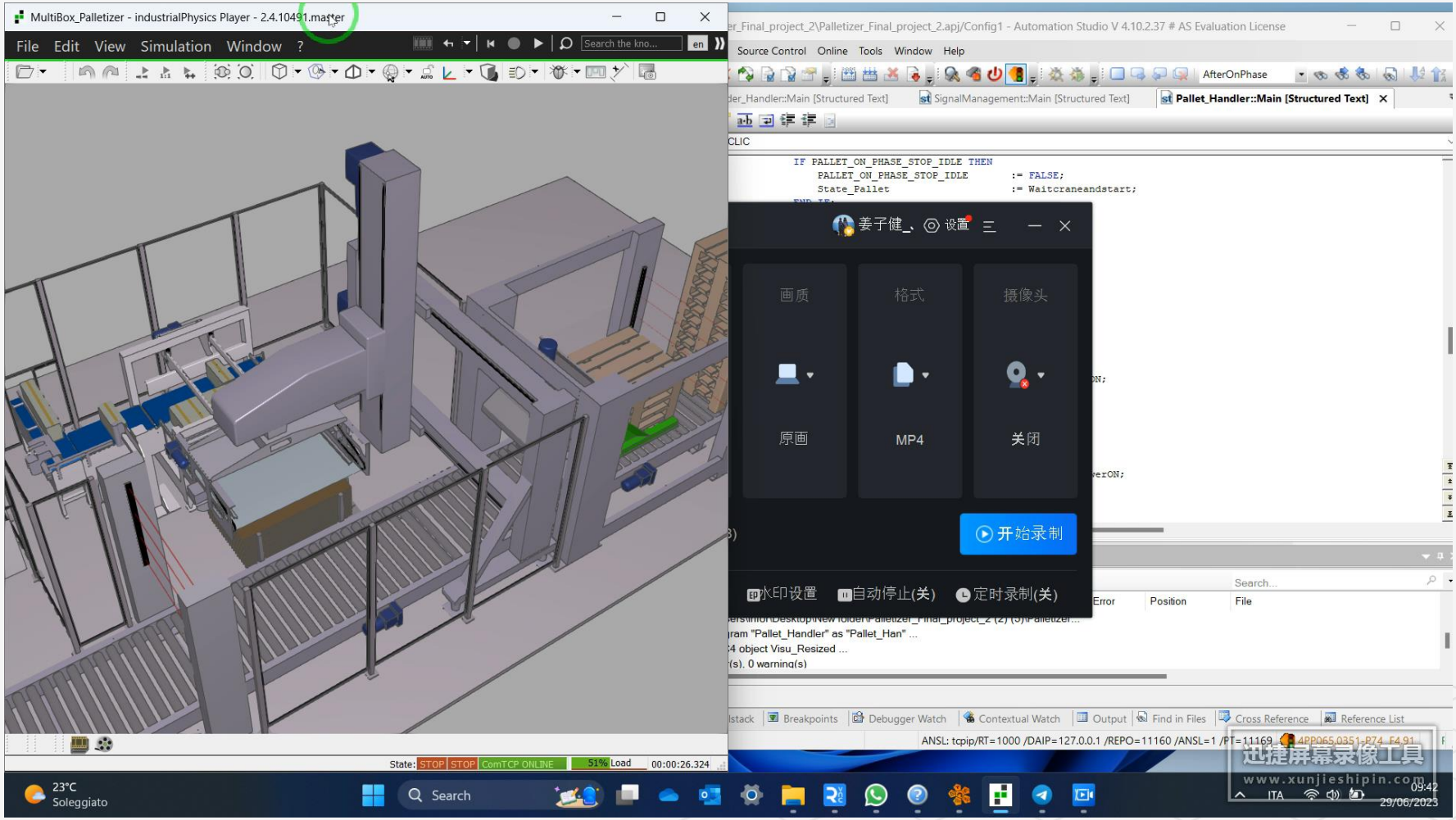
Signal and Error Management Simulation (On-Phase Stop)



Signal and Error Management Simulation (Immediate Stop)



Signal and Error Management Simulation (Emergency Stop)



Thanks for your Attention