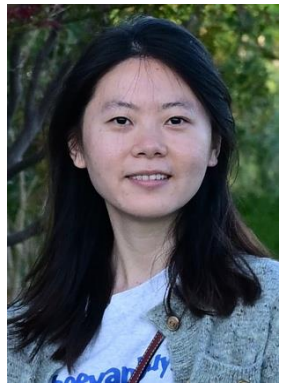
Property-based Testing of Wafer Handling Robots via Model-based Planning and Fuzzing

Ruiyang Xu, Jingjing Liang, Guoyue Zheng, Geguang Pu, Ting Su

East China Normal University

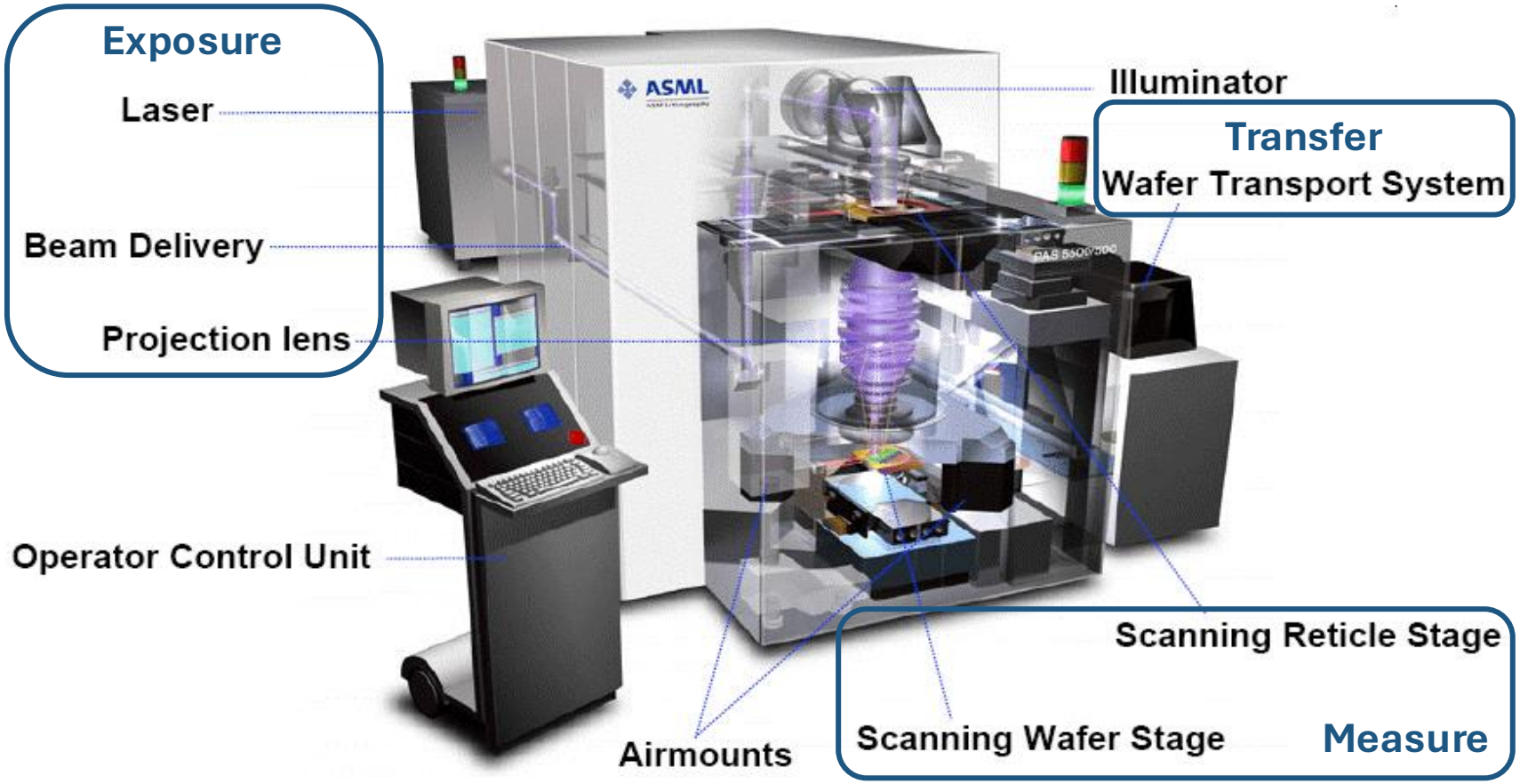
ASE 2026

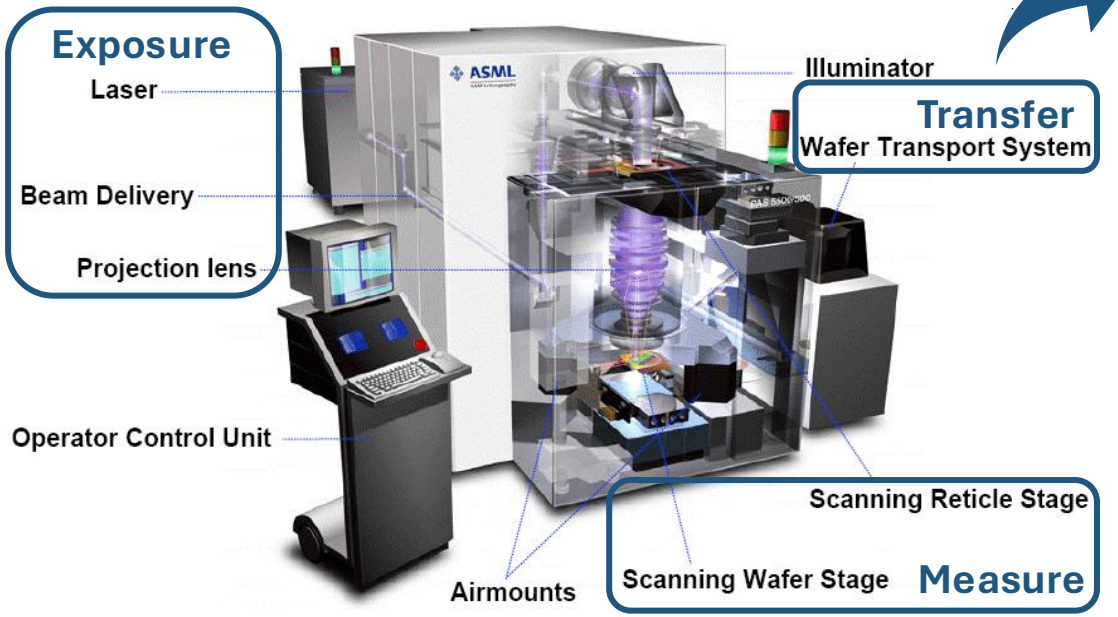



Background: Lithography Equipment

- Lithography equipment **prints chip circuit** design onto silicon wafers.
- 3 Core Subsystems: **(1) Transfer**, (2) Measure, and (3) Exposure



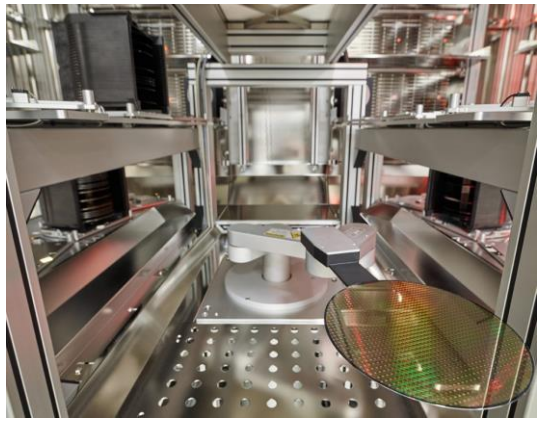
Background: Wafer Handling Robots(WHRs)



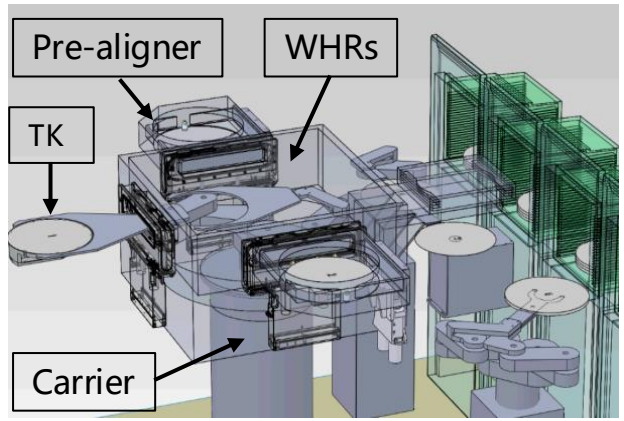
Lithography equipment (ASML EUV)

Wafer Handling System

Wafer handling robots are used for **transferring wafers** at high speed and precision in a confined workspace



Wafer handling system

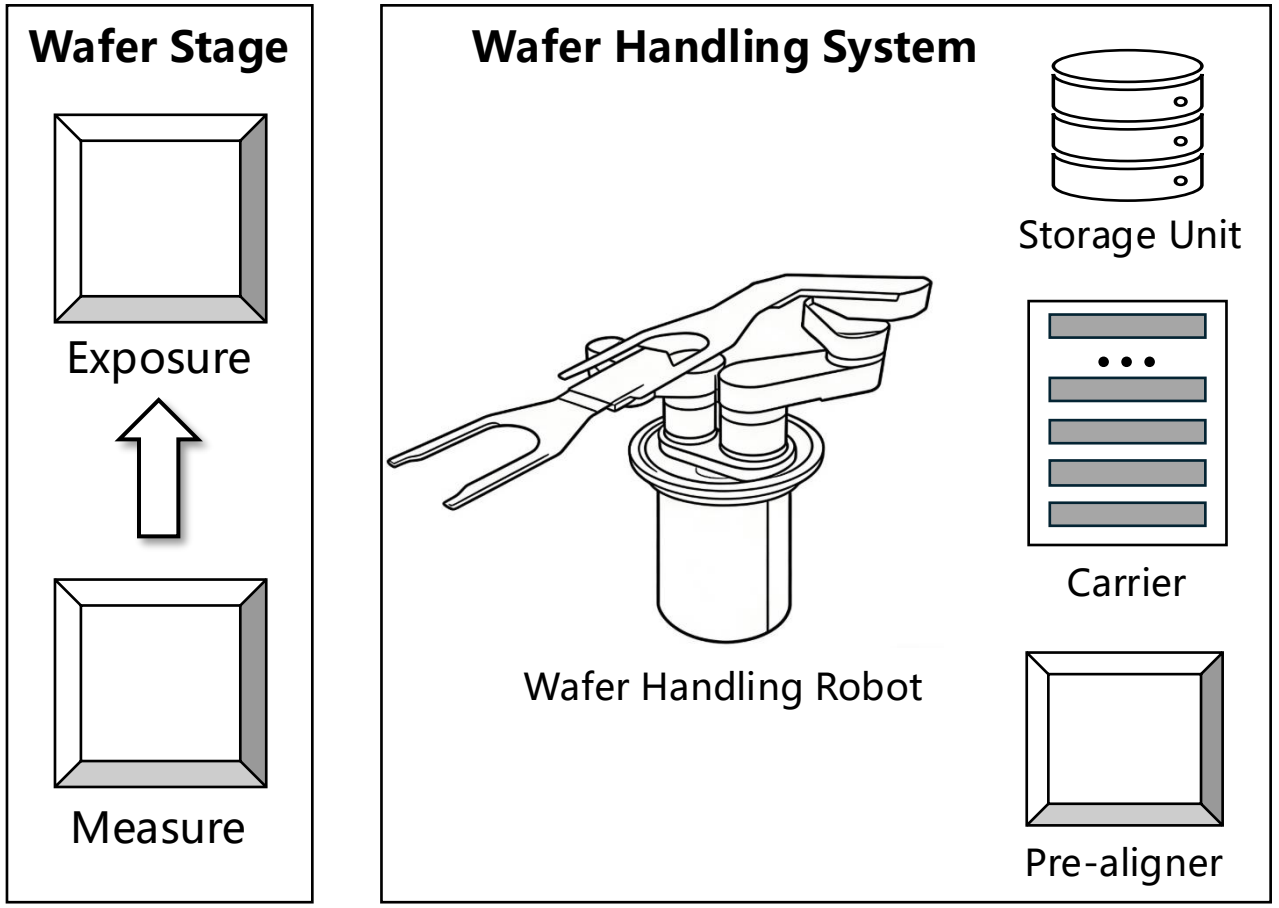


3D Structural Model

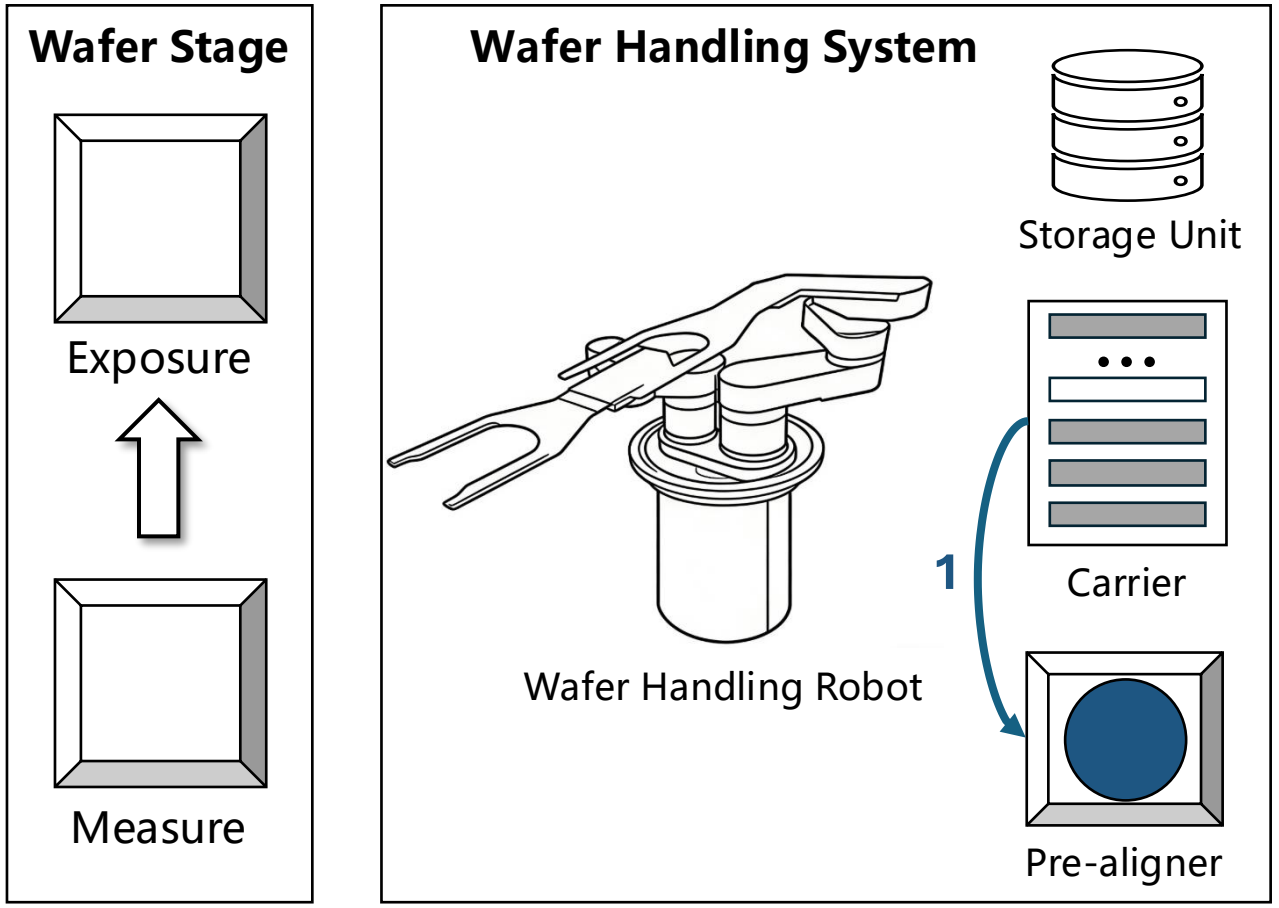
ADD 1

ADD 1

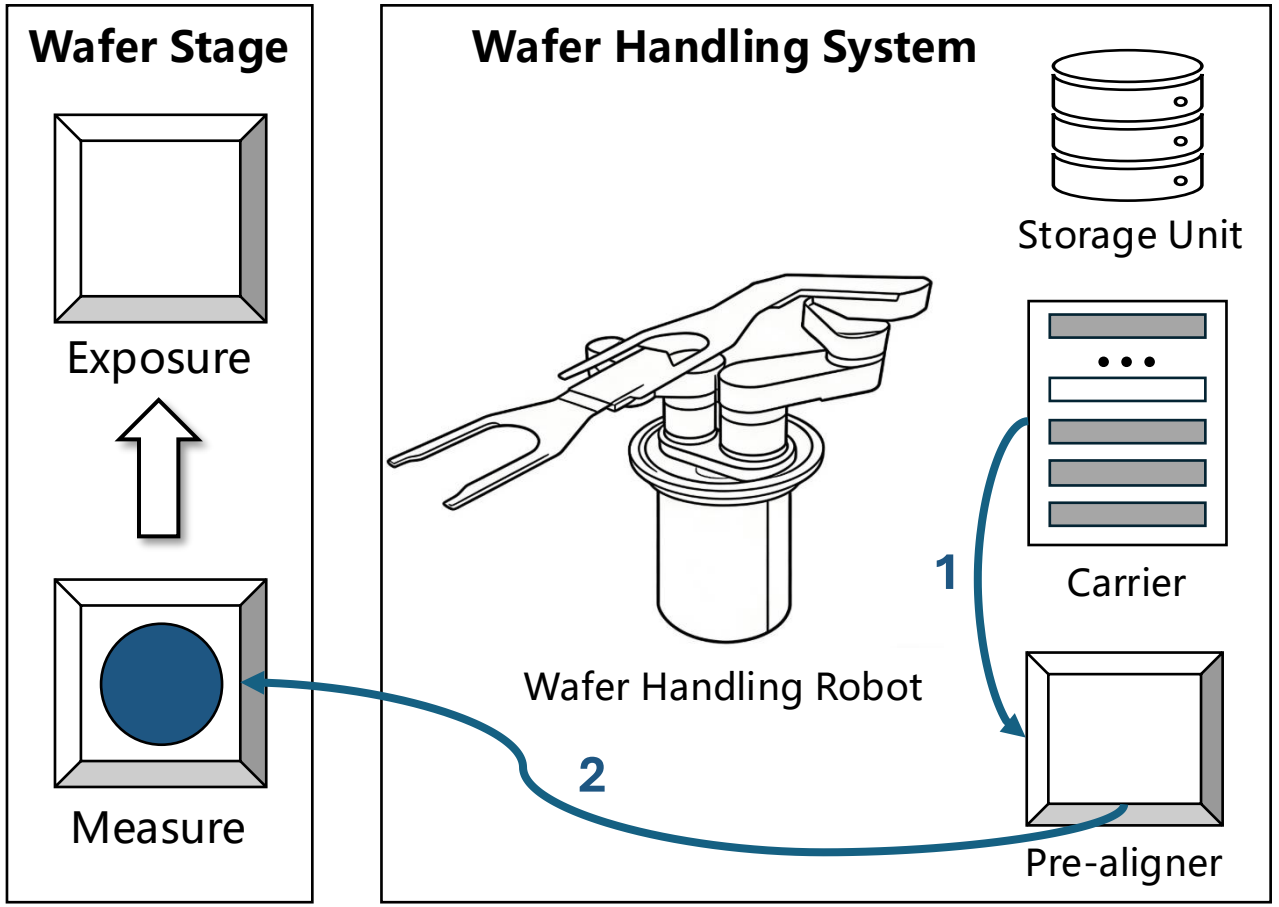
Background: Wafer Transfer Flow



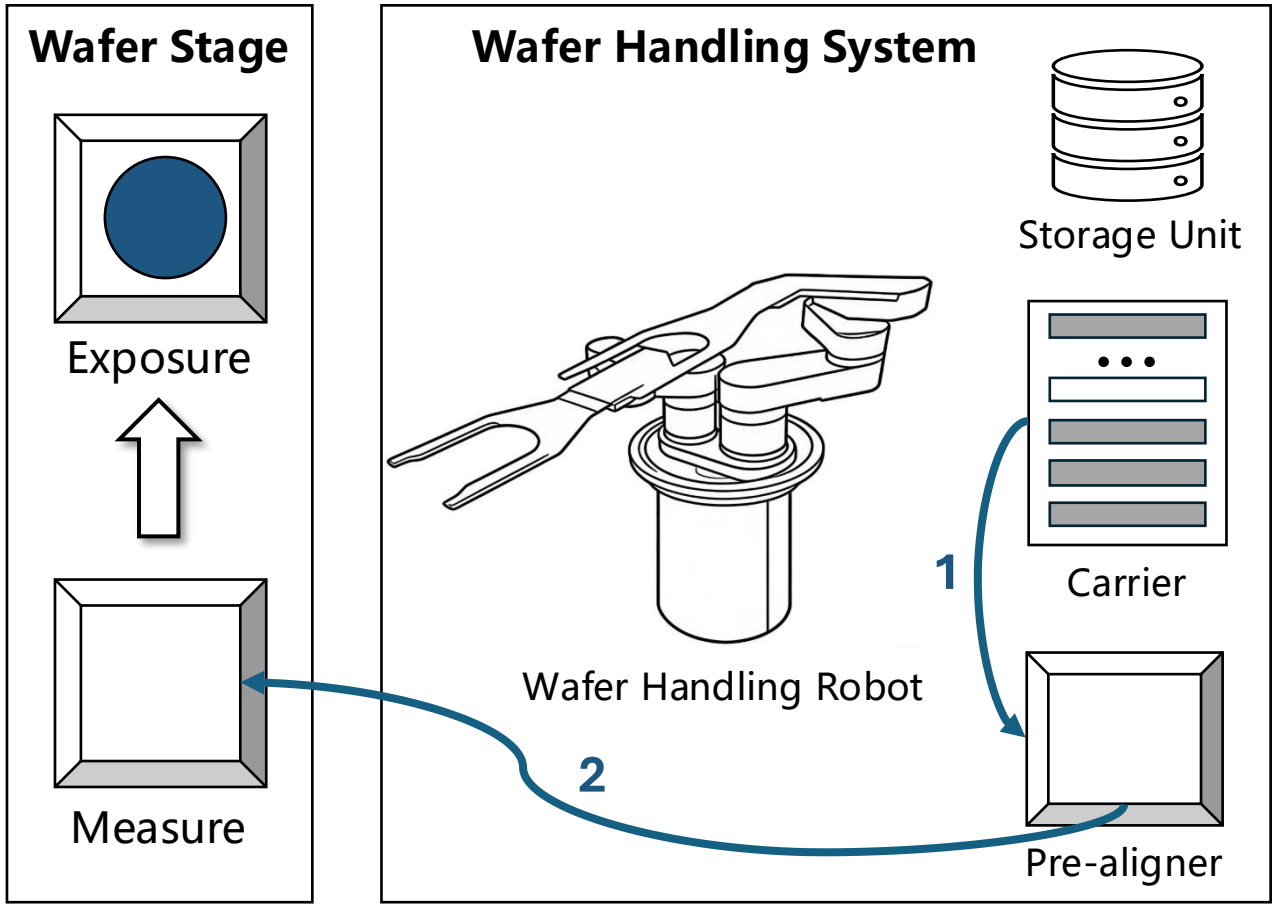
Background: Wafer Transfer Flow



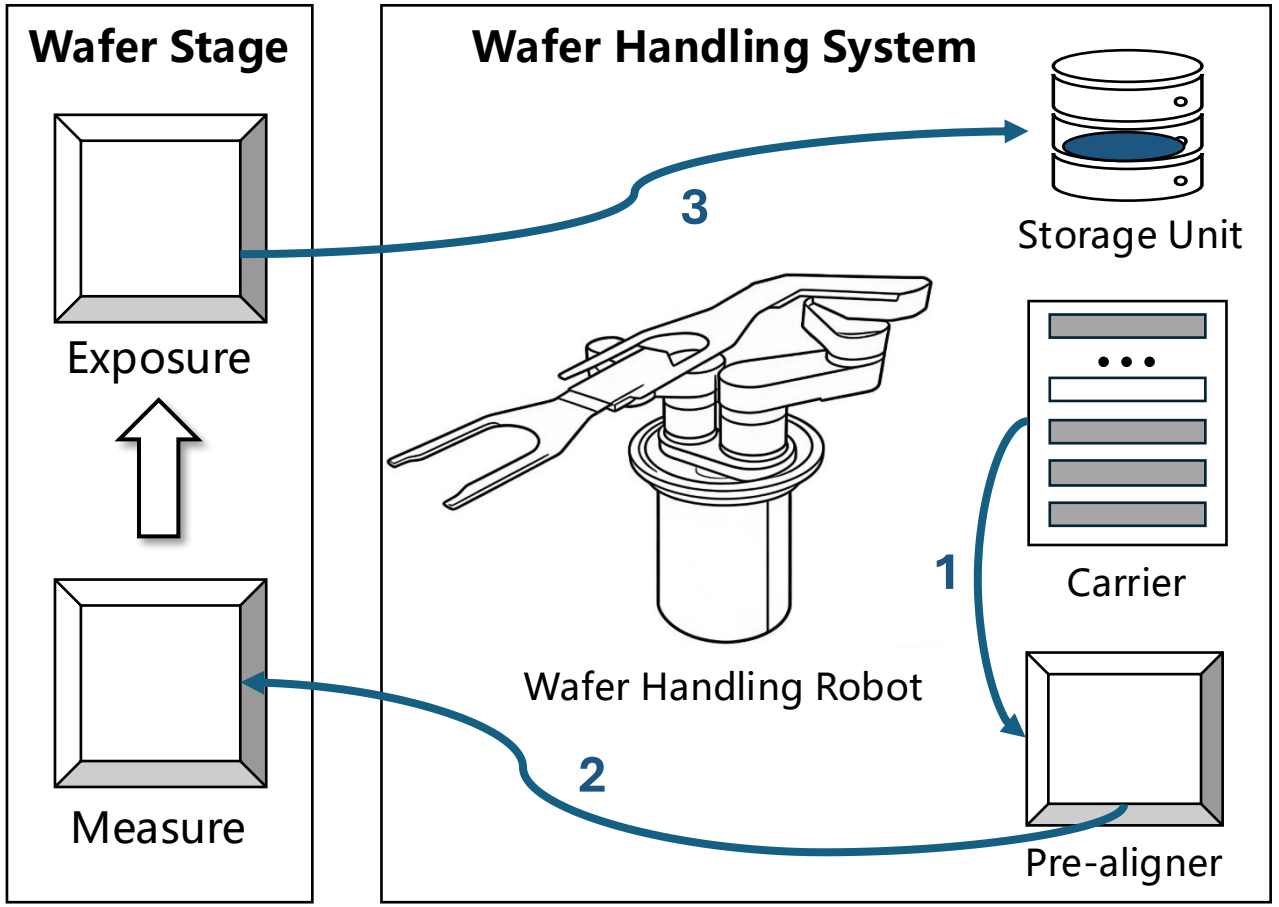
Background: Wafer Transfer Flow



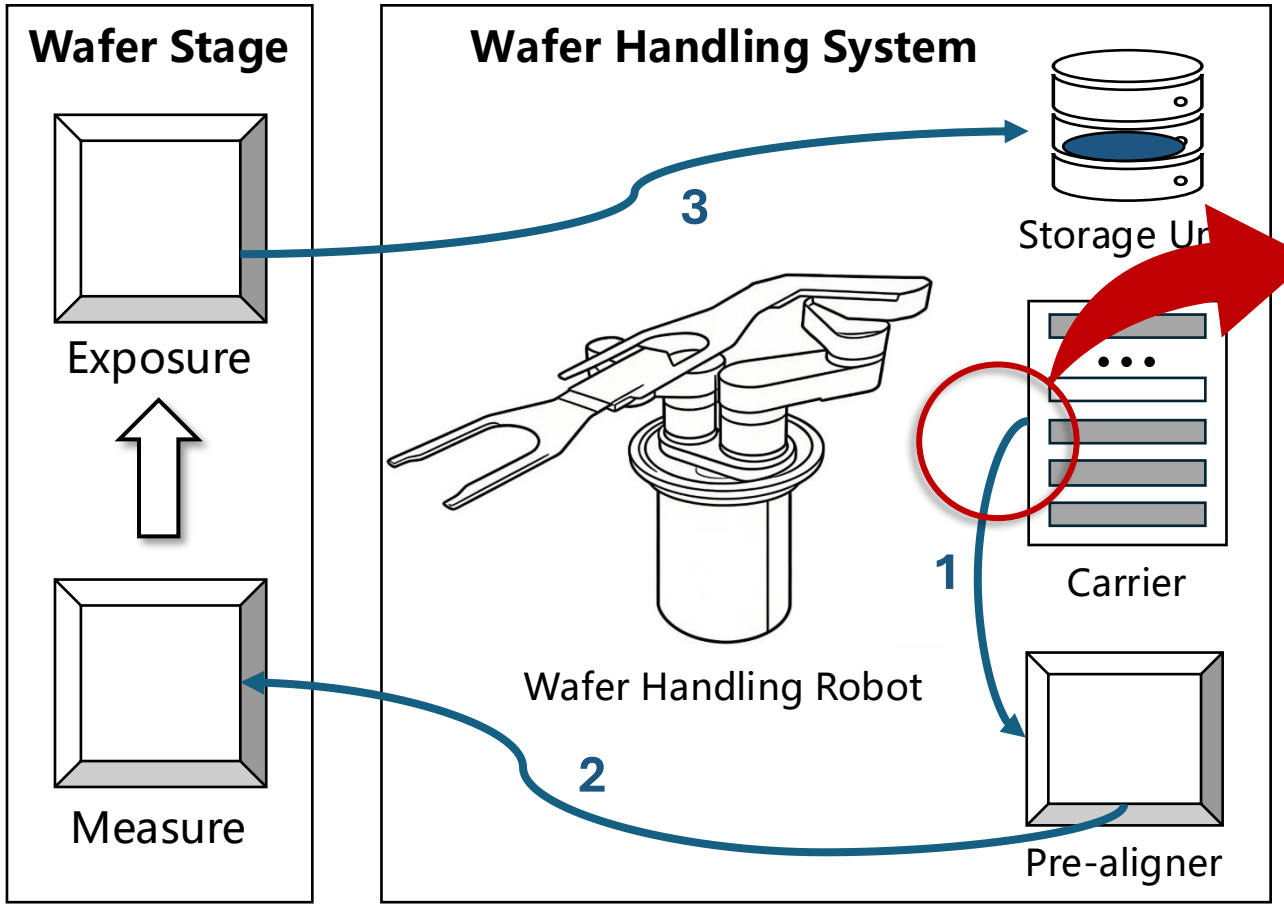
Background: Wafer Transfer Flow

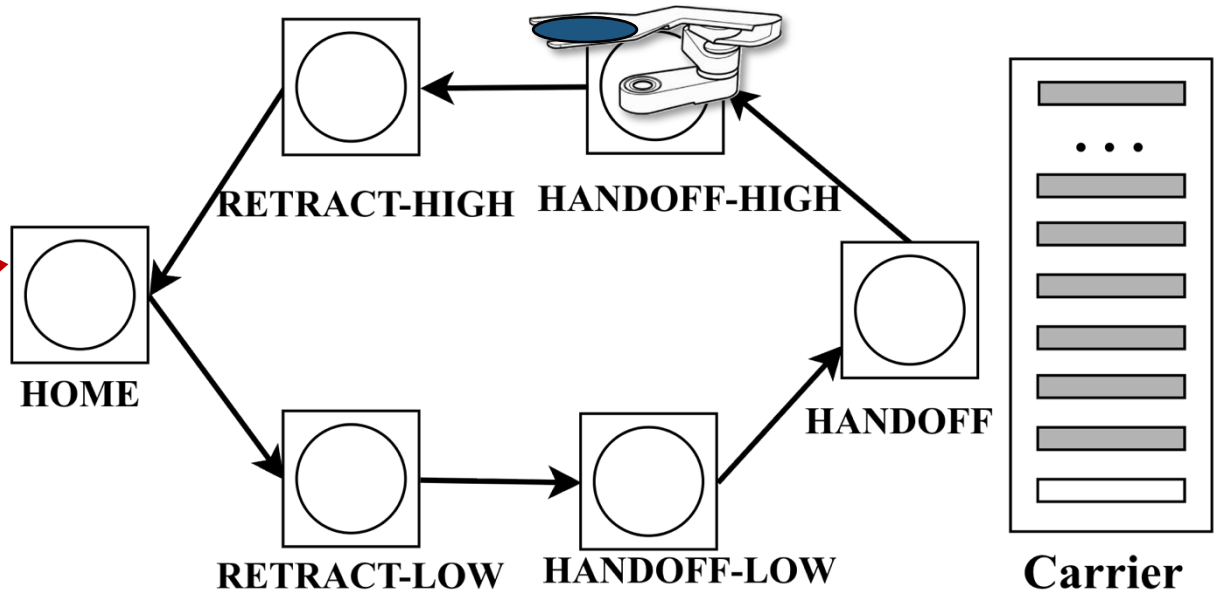


Background: Wafer Transfer Flow



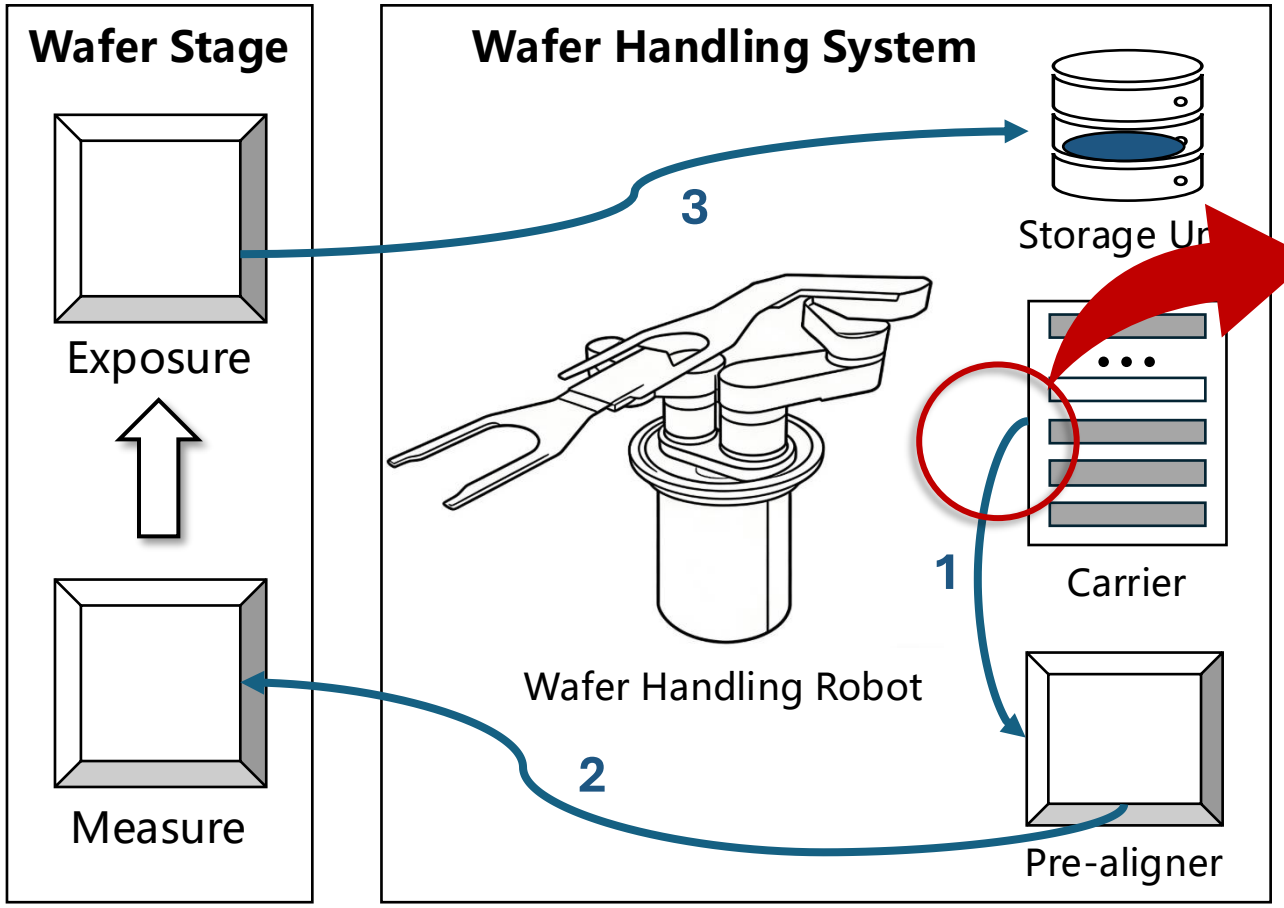
Background: Wafer Transfer Flow

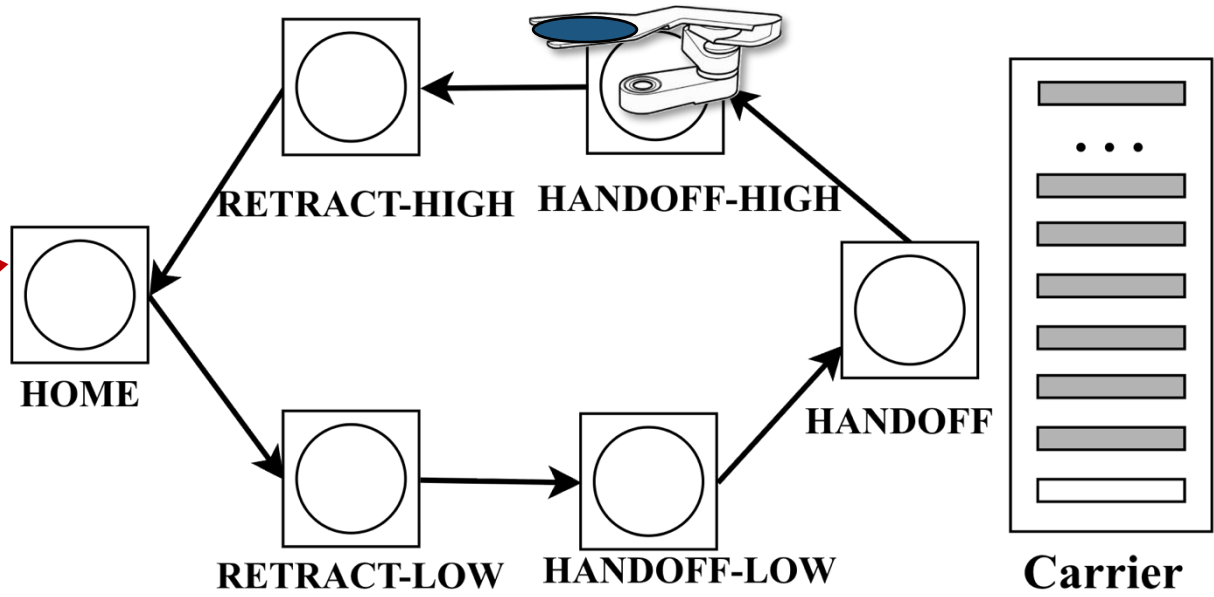




WHR is getting wafer from the carrier station by station

Background: Wafer Transfer Flow



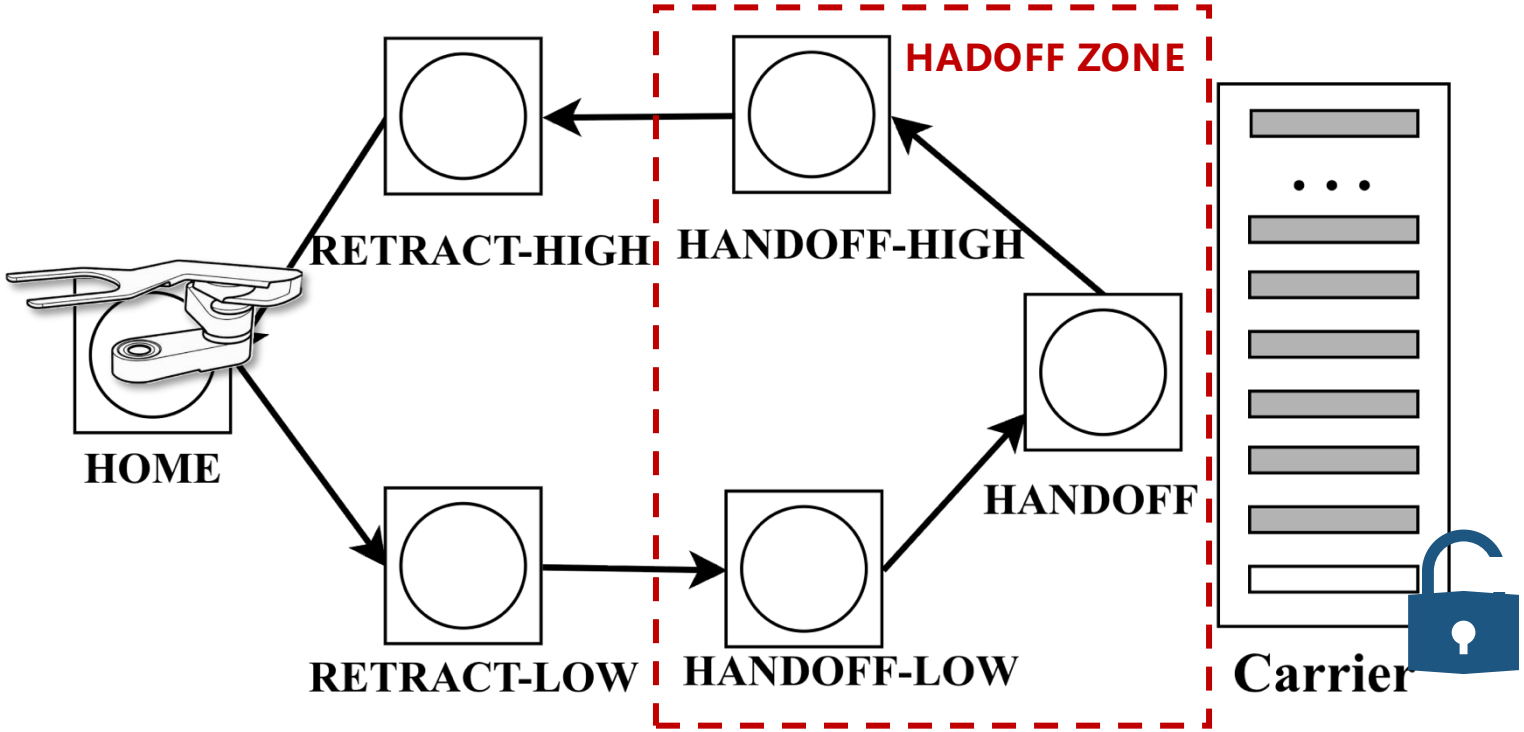


WHR is getting wafer from the carrier station by station

GOAL: Validating the Reliability of Wafer Handling Robots (WHRs)

Observation

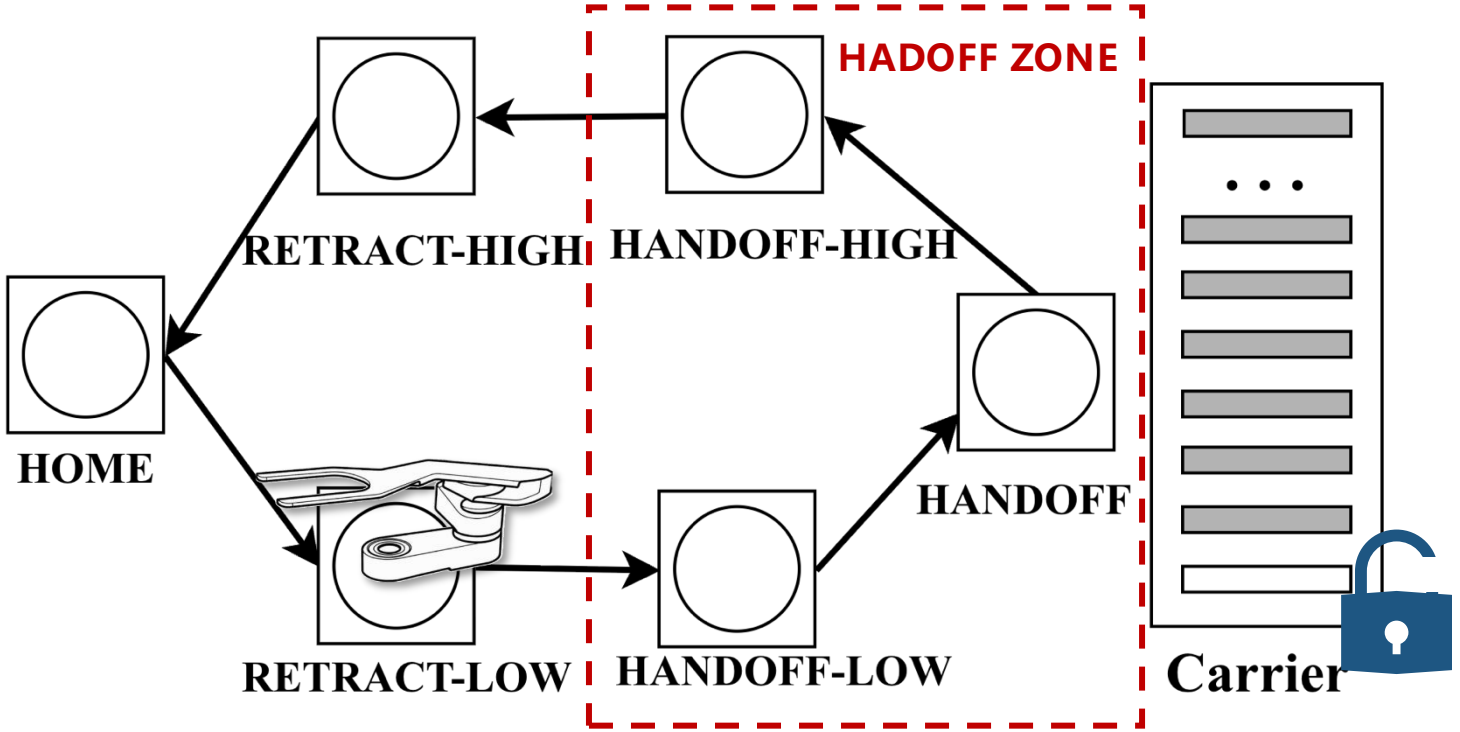
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone*" (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

Observation

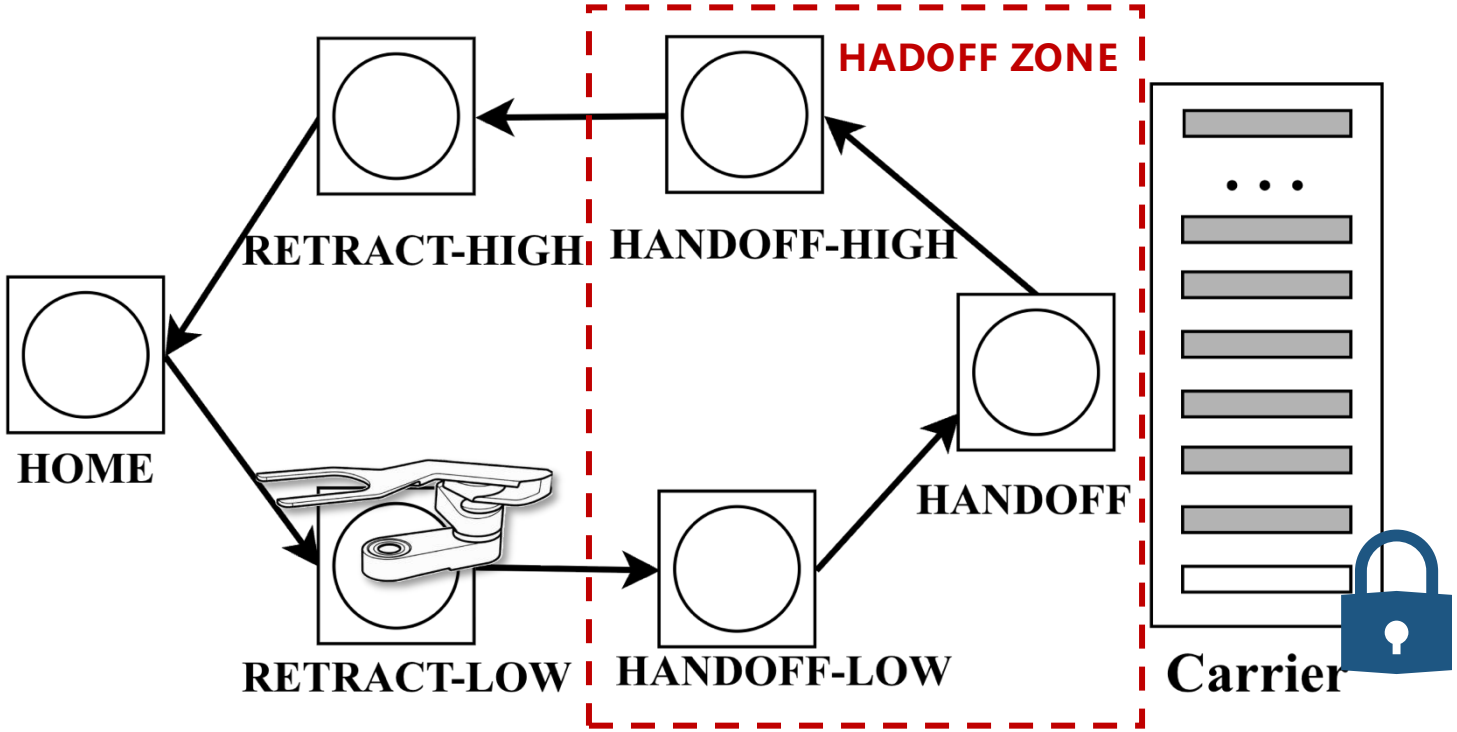
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone*" (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

Observation

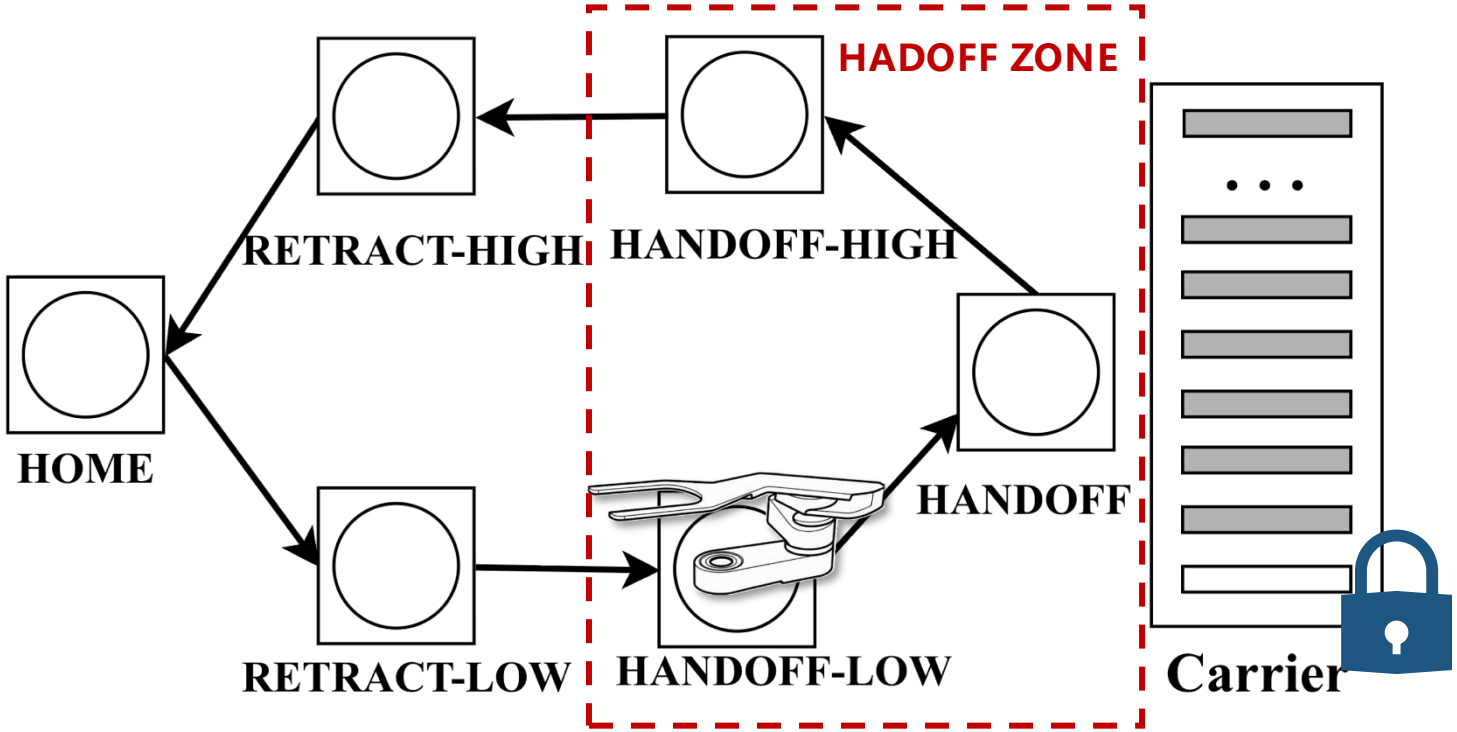
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone*" (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

Observation

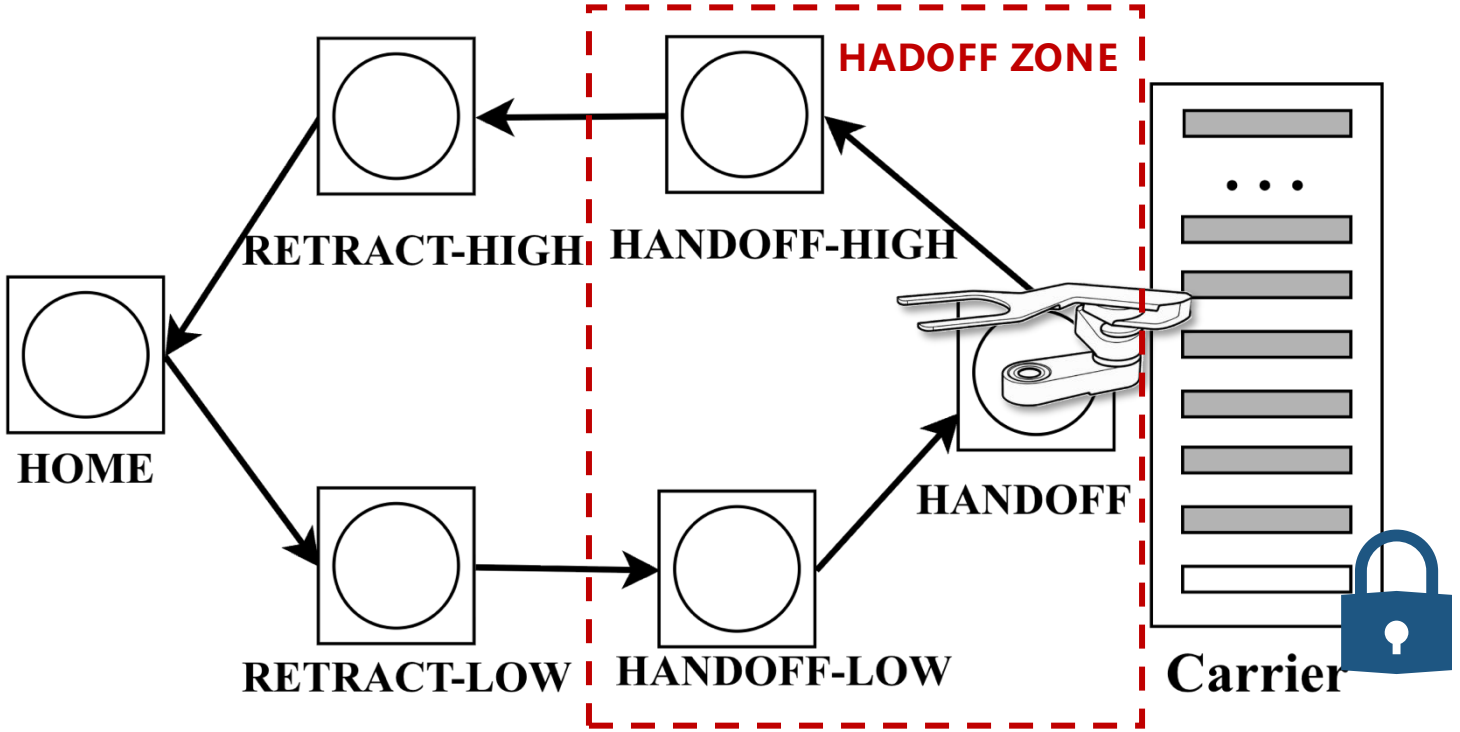
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone"* (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

Observation

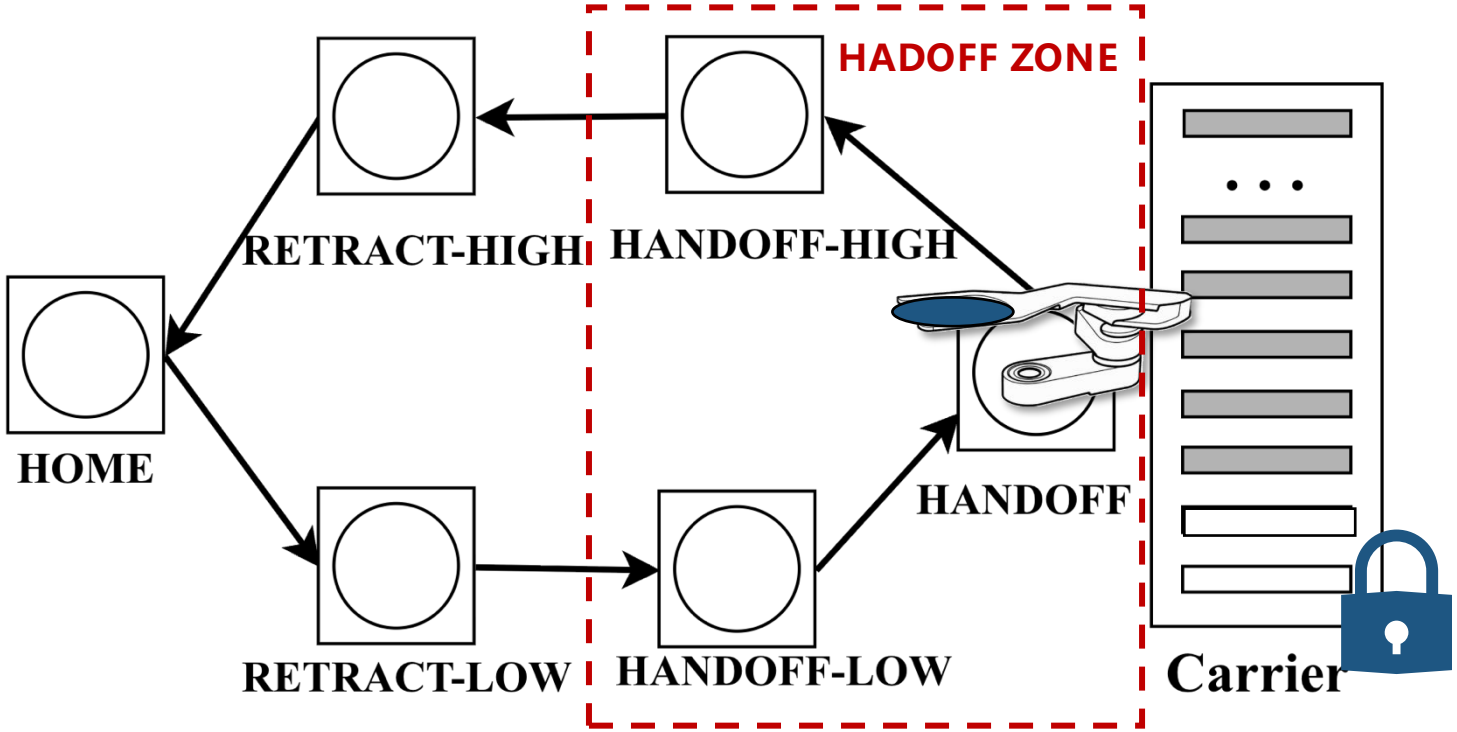
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone*" (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

Observation

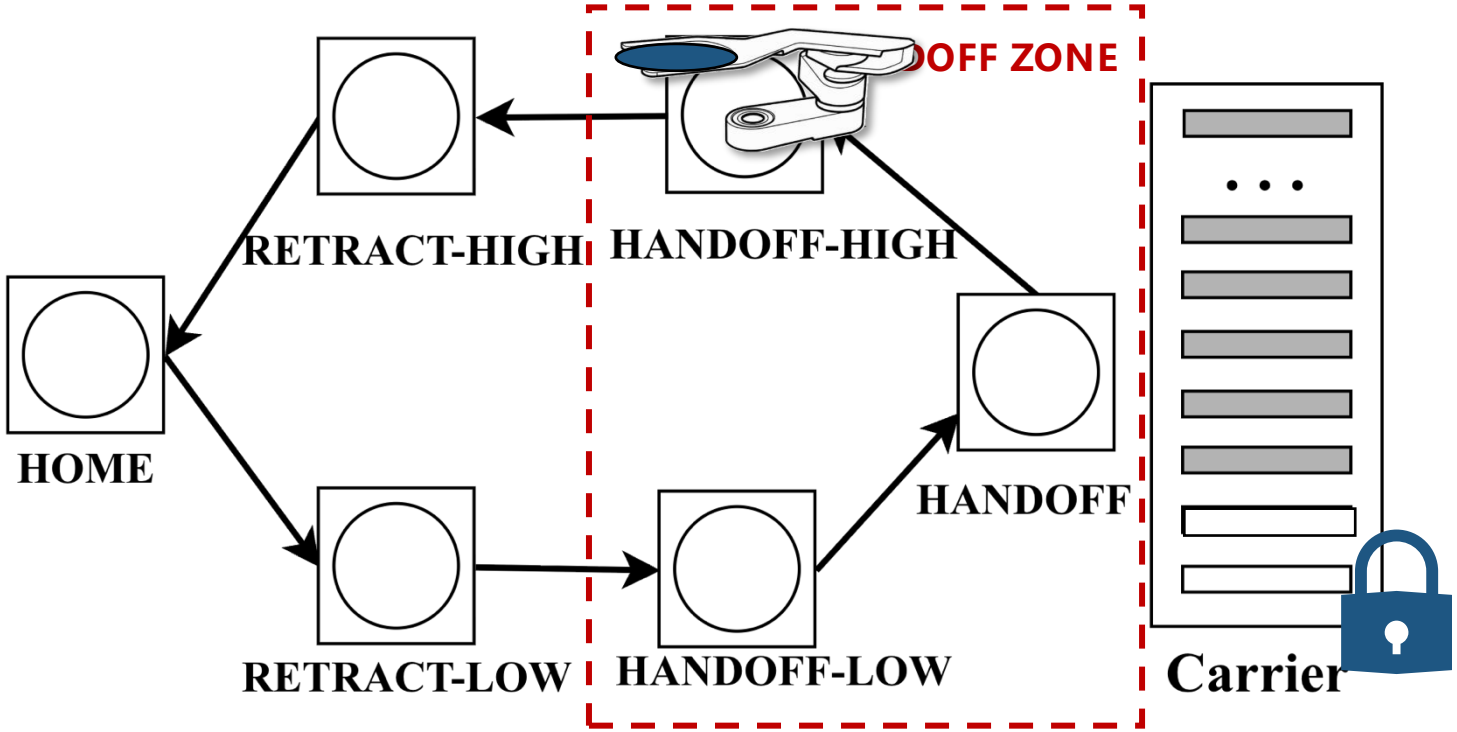
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone"* (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

Observation

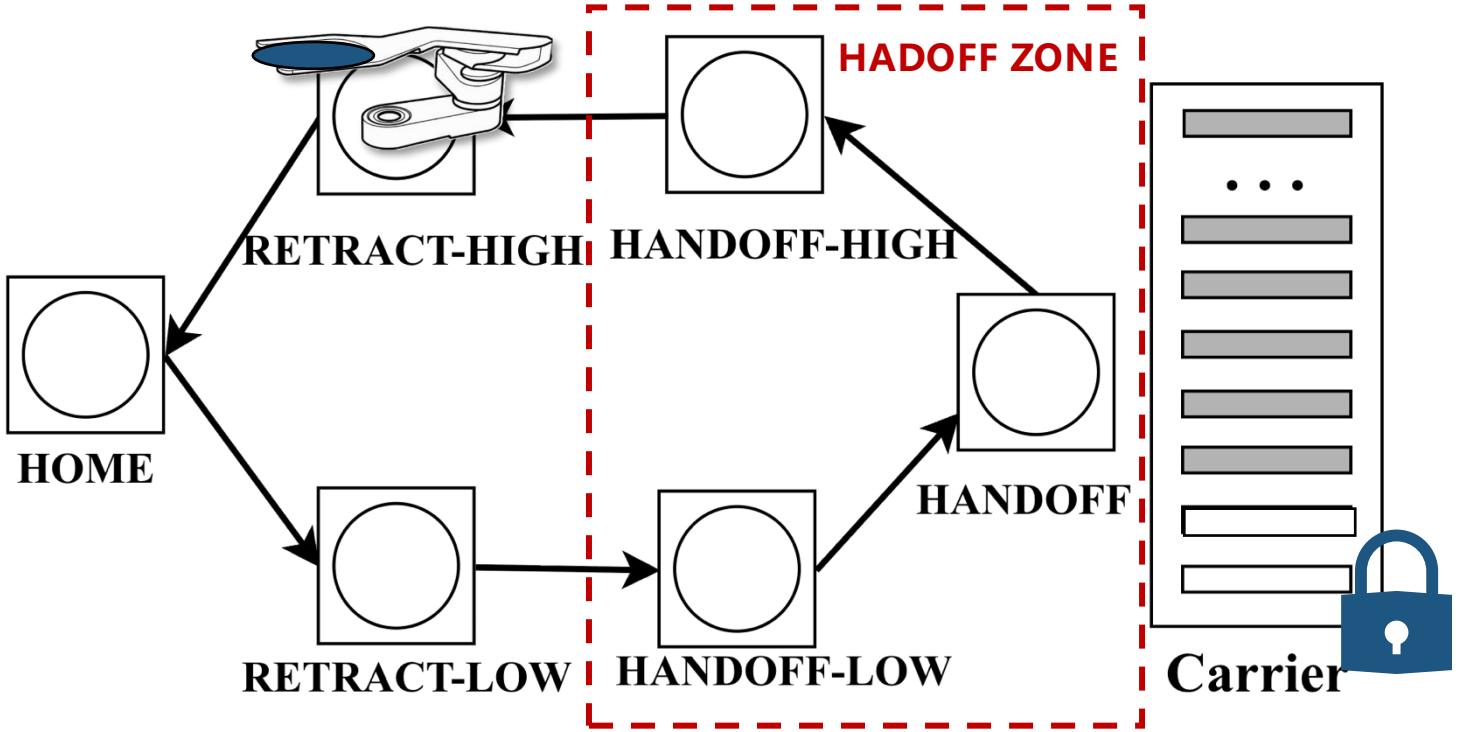
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone*" (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

Observation

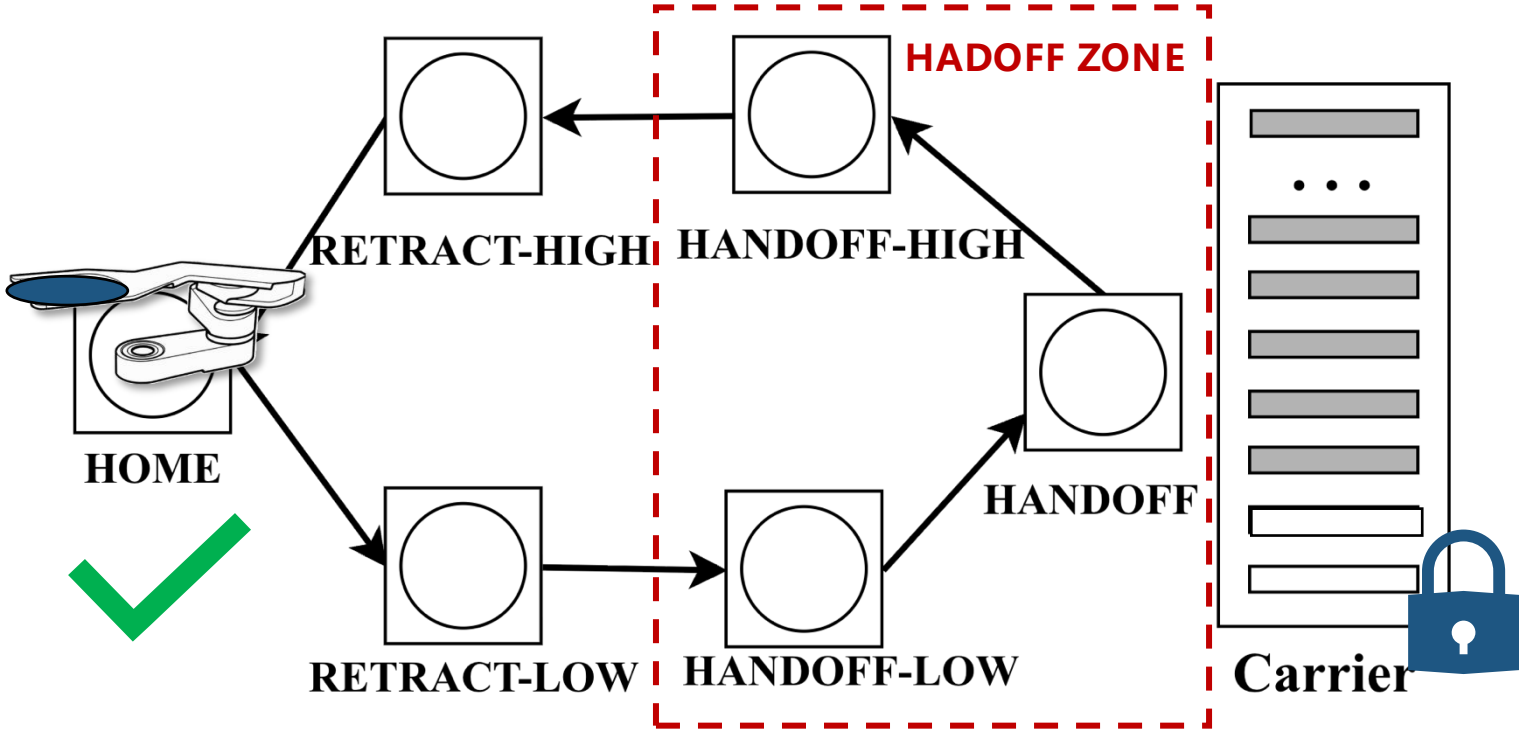
- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone*" (red dash box) to ensure stable alignement.



WHR is getting wafer from the carrier station by station

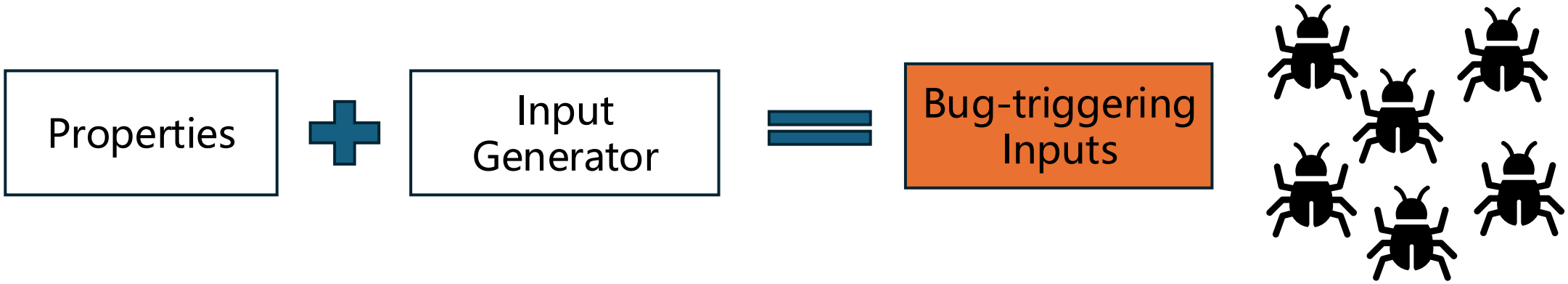
Observation

- **Observation:** WHRs are expected to respect a number of important system requirements.
- **Example:** The carrier "must *be locked when the WHR enters and stays in the handoff zone*" (red dash box) to ensure stable alignement.



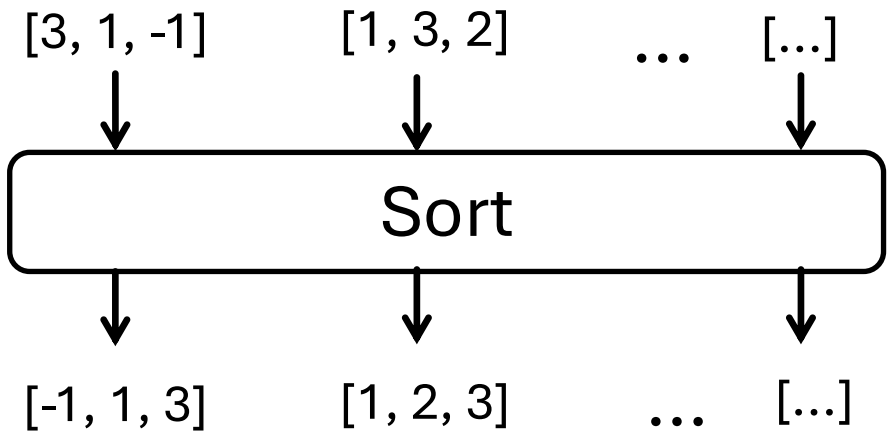
WHR is getting wafer from the carrier station by station

Idea: Property-based Testing (PBT)



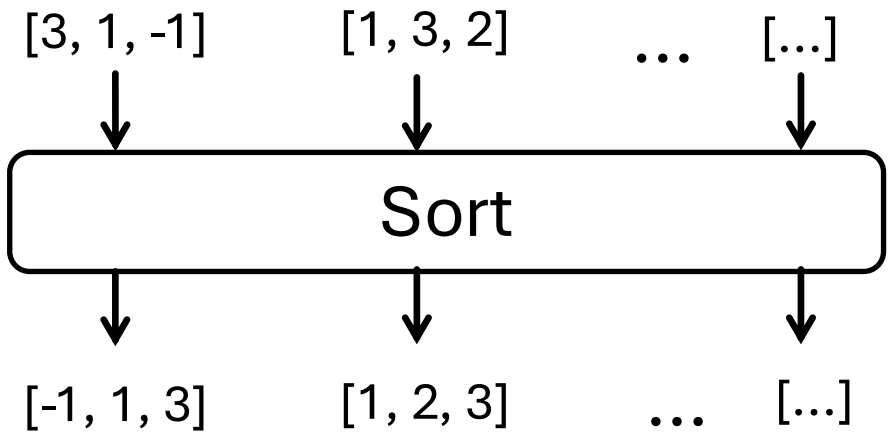
Koen Claessen and John Hughes. QuickCheck: a lightweight tool for random testing of Haskell programs. ICFP 2000

Example: PBT for a Sort Function

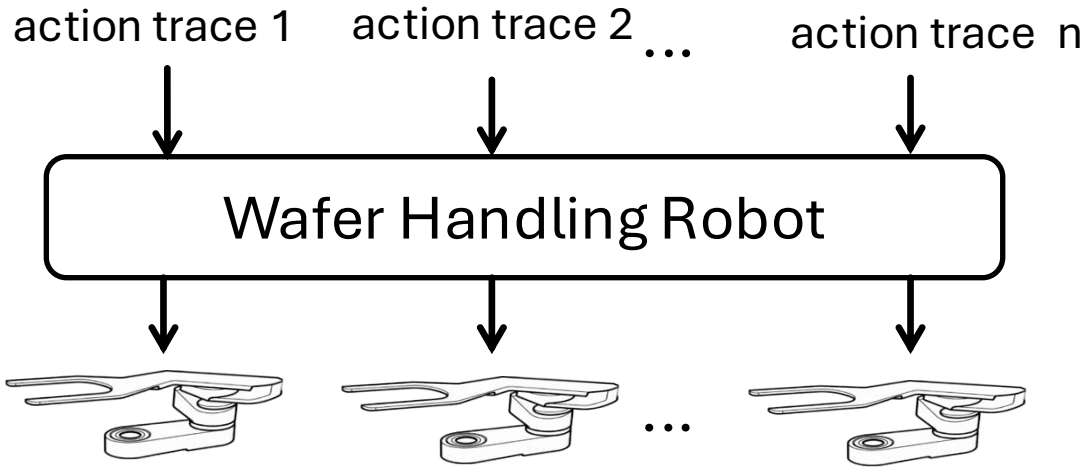


Property:
The elements of output list should be ascending.

Property-based Testing for WHRs

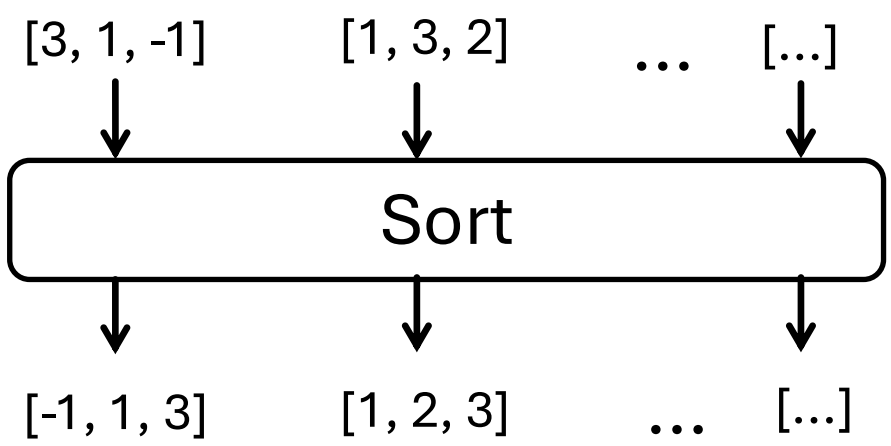


Property:
The elements of output list should be ascending.

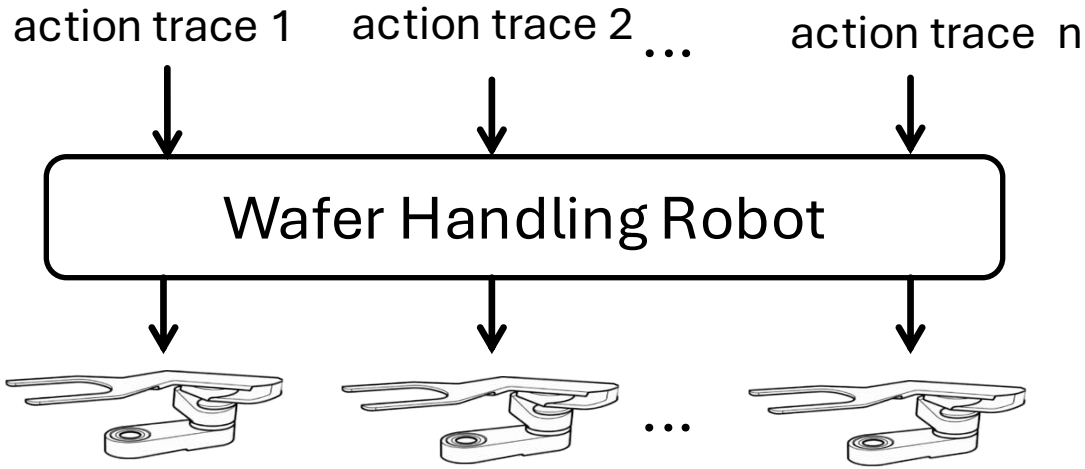


Property:
System requirements that a WHR should respect

Technical Challenges of PBT for WHRs



Property:
The elements of output list should be ascending.



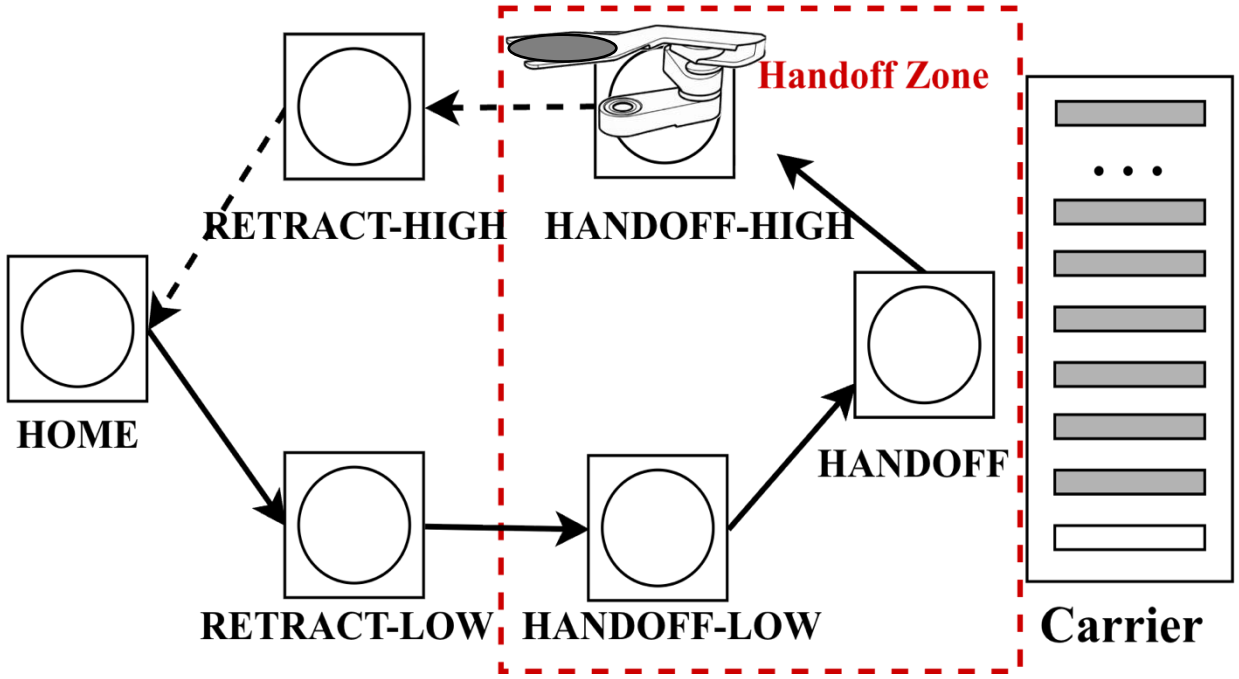
Property:
System requirements that a WHR should respect

How to effectively generate test input for WHRs?

Challenges of Input Generation

- **Challenge 1 (Action Dependency)**

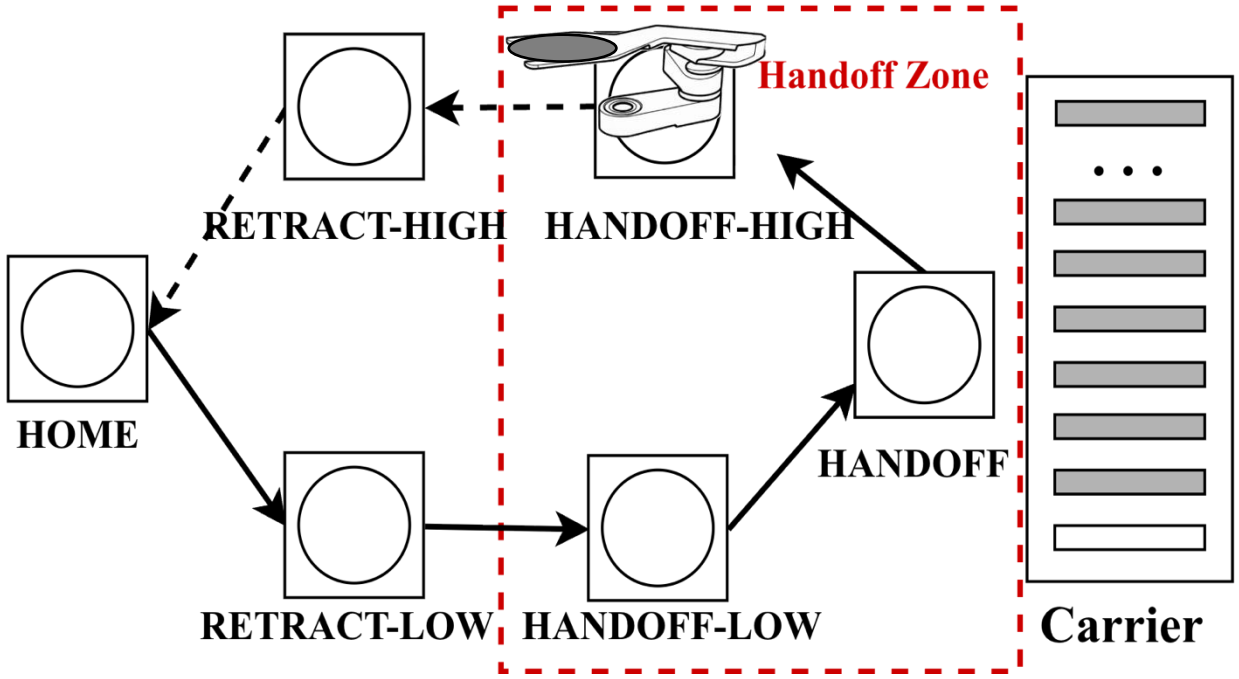
Actions of WHRs have complex dependencies



Challenges of Input Generation

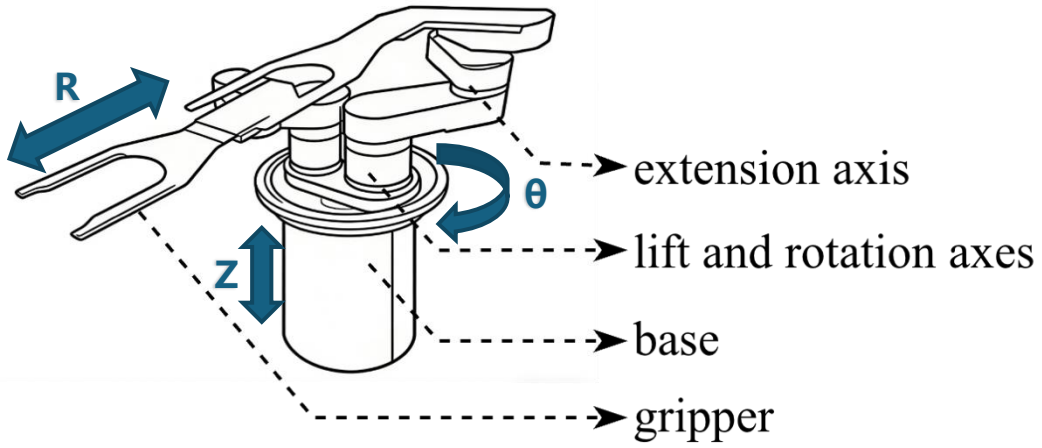
➢ **Challenge 1 (Action Dependency)**

Actions of WHRs have complex dependencies



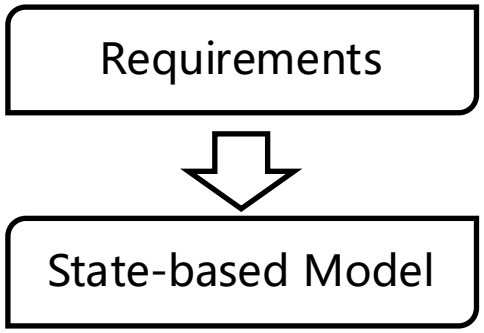
➢ **Challenge 2 (State Space Explosion)**

State space of a WHR system is huge



WHRs have continuous parameters (lift height, rotation angle) and dynamic environments (carrier locked/unlocked)

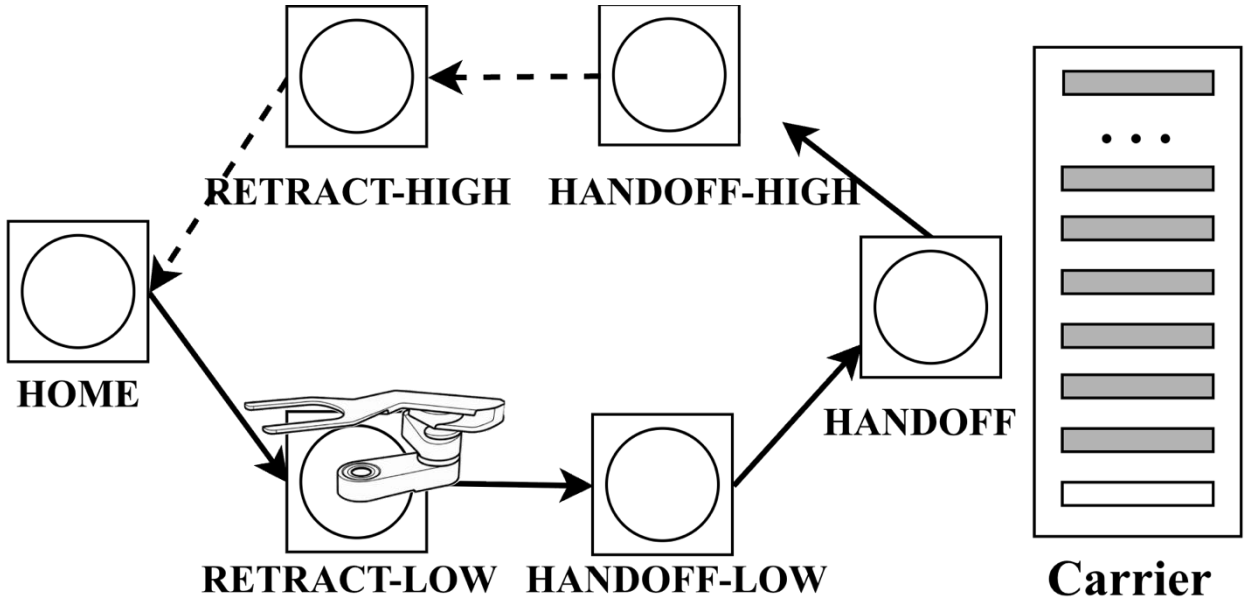
Approach: Leveraging the Model



Example of the Model

Requirements

State-based Model

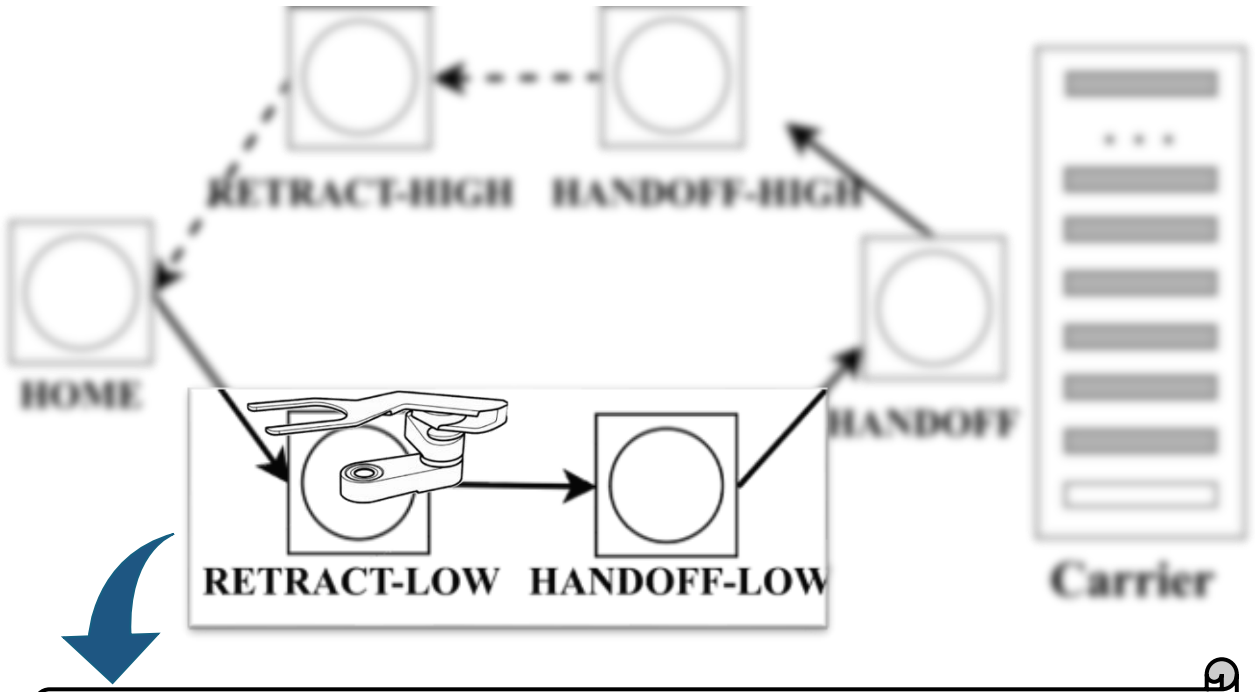


Example of the Model

Requirements

⇩

State-based Model



MOVE: FROM RETRACT-LOW TO HANDOFF-HIGH

GUARDS:

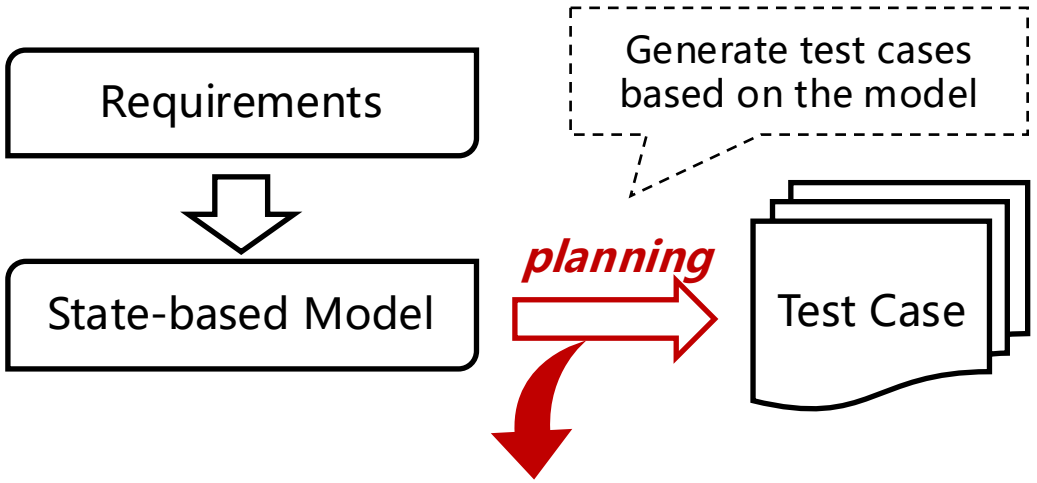
1. CurrentStation = RETRACT-LOW
2. CarrierLocked = TRUE

EFFECT:

1. CurrentStation = HANDOFF-LOW

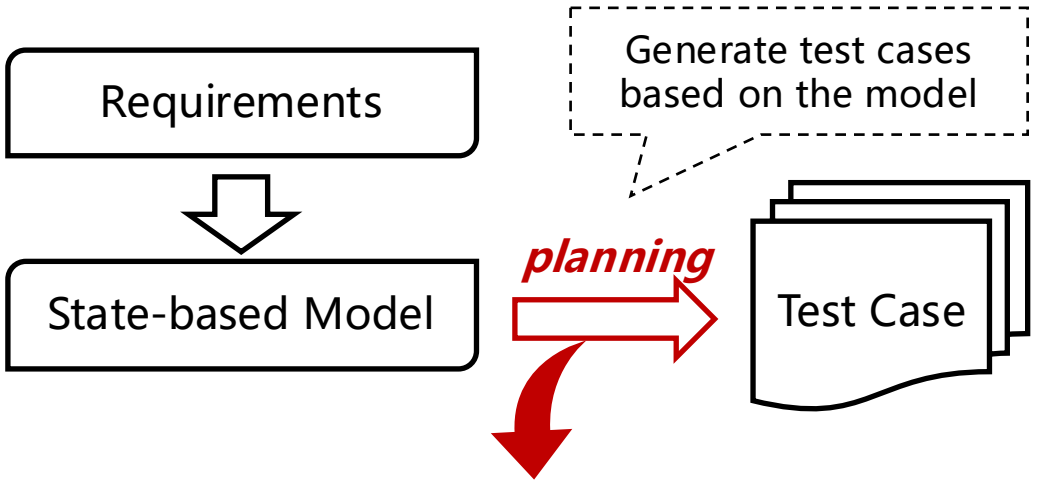
(For details of this model, please refer to our paper)

Approach: Model-based Planning



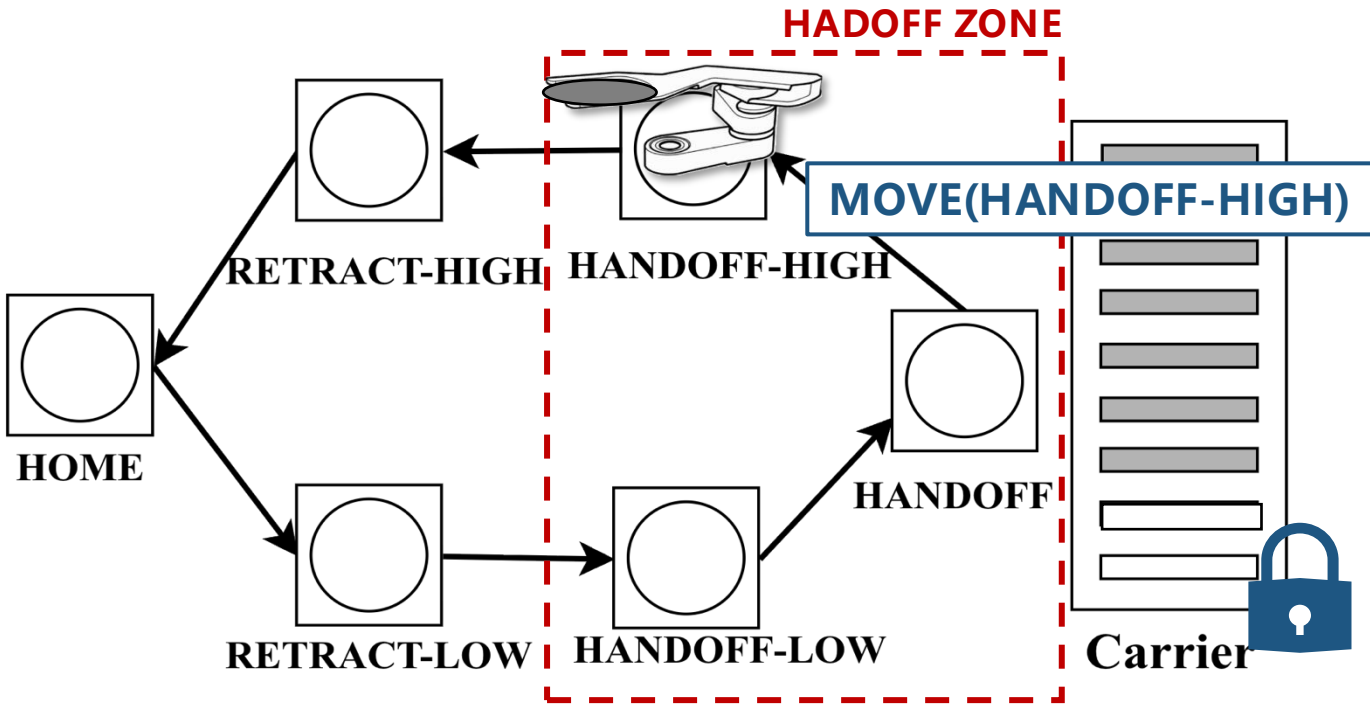
Mitigate the Challenge 1 (Action Dependency)

Example of Model-based Planning



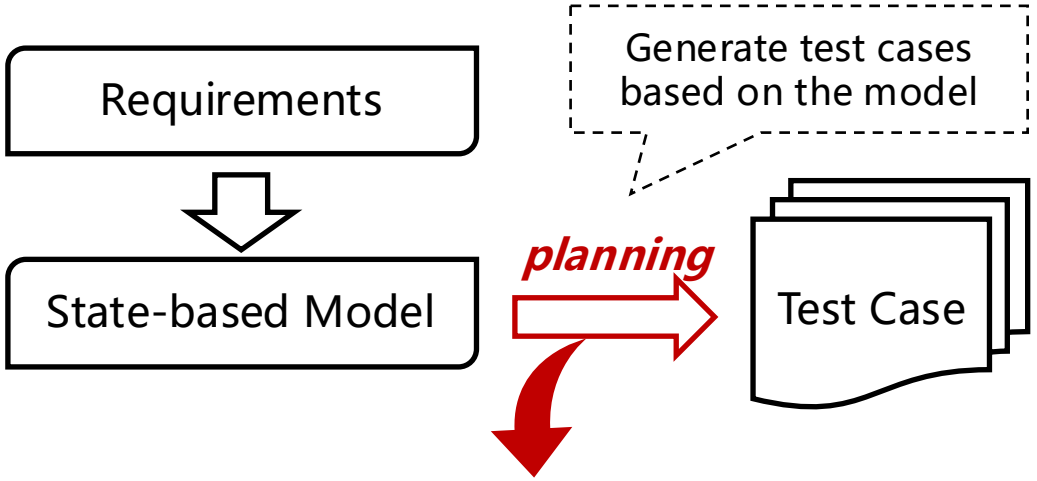
Mitigate the Challenge 1 (Action Dependency)

Planned Test Cases



WHR is getting wafer from the carrier station by station

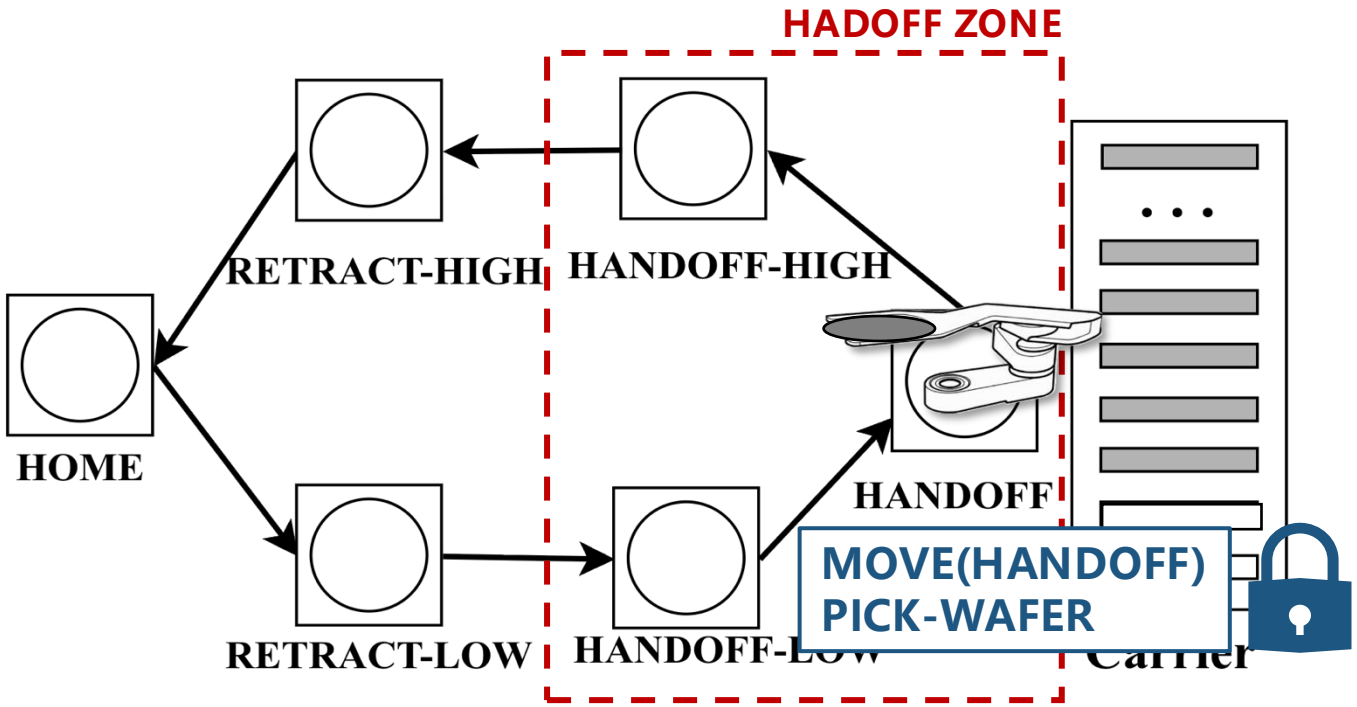
Example of Model-based Planning



Mitigate the Challenge 1 (Action Dependency)

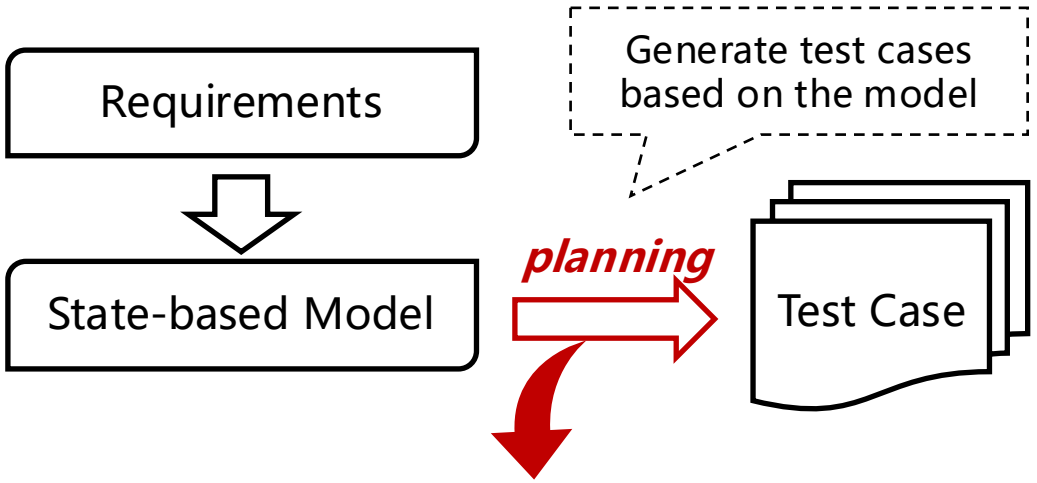
Planned Test Cases

MOVE(HANDOFF-HIGH)



WHR is getting wafer from the carrier station by station

Example of Model-based Planning

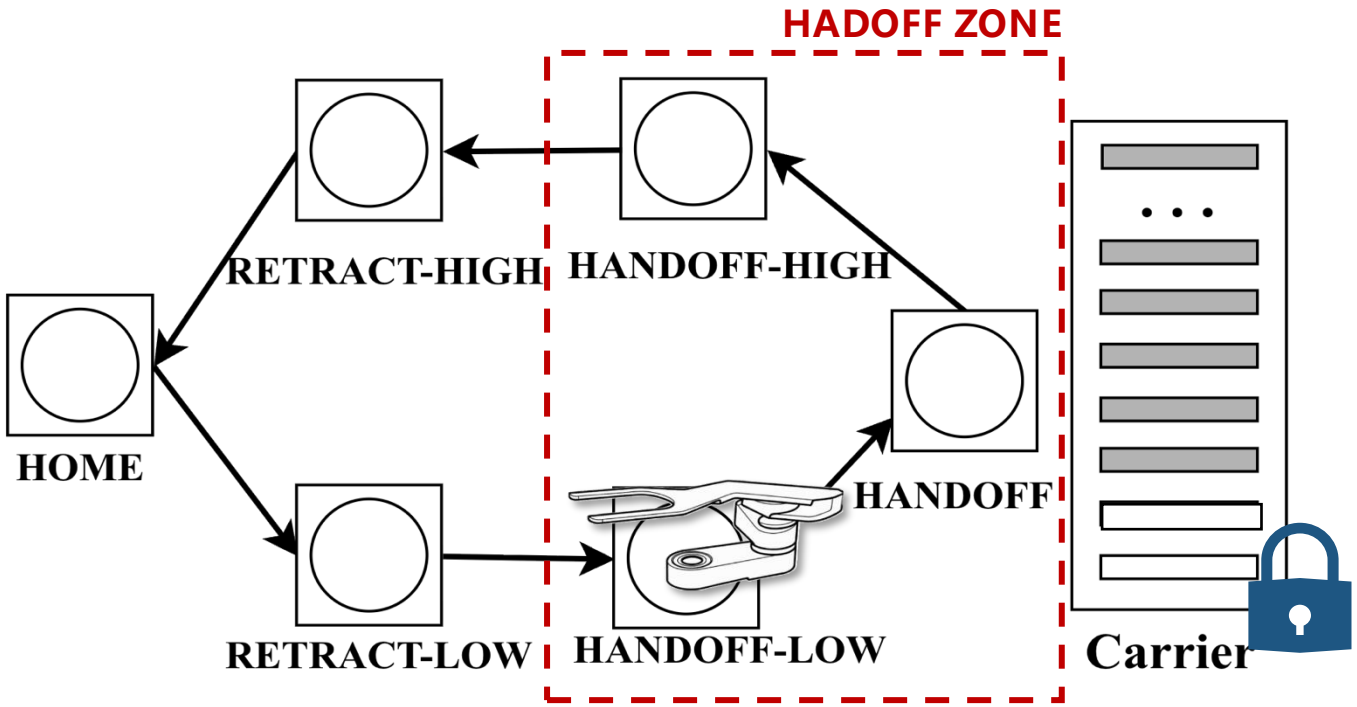


Mitigate the Challenge 1 (Action Dependency)

Planned Test Cases

MOVE(HANDOFF-HIGH)

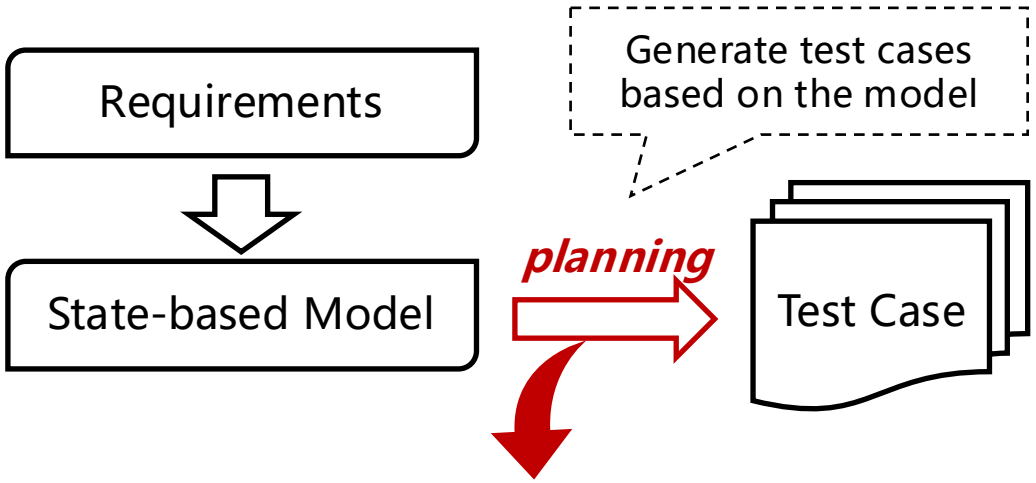
MOVE(HANDOFF)
PICK-WAFER



WI station by station

MOVE(HANDOFF-LOW)
LOCK-CARRIER

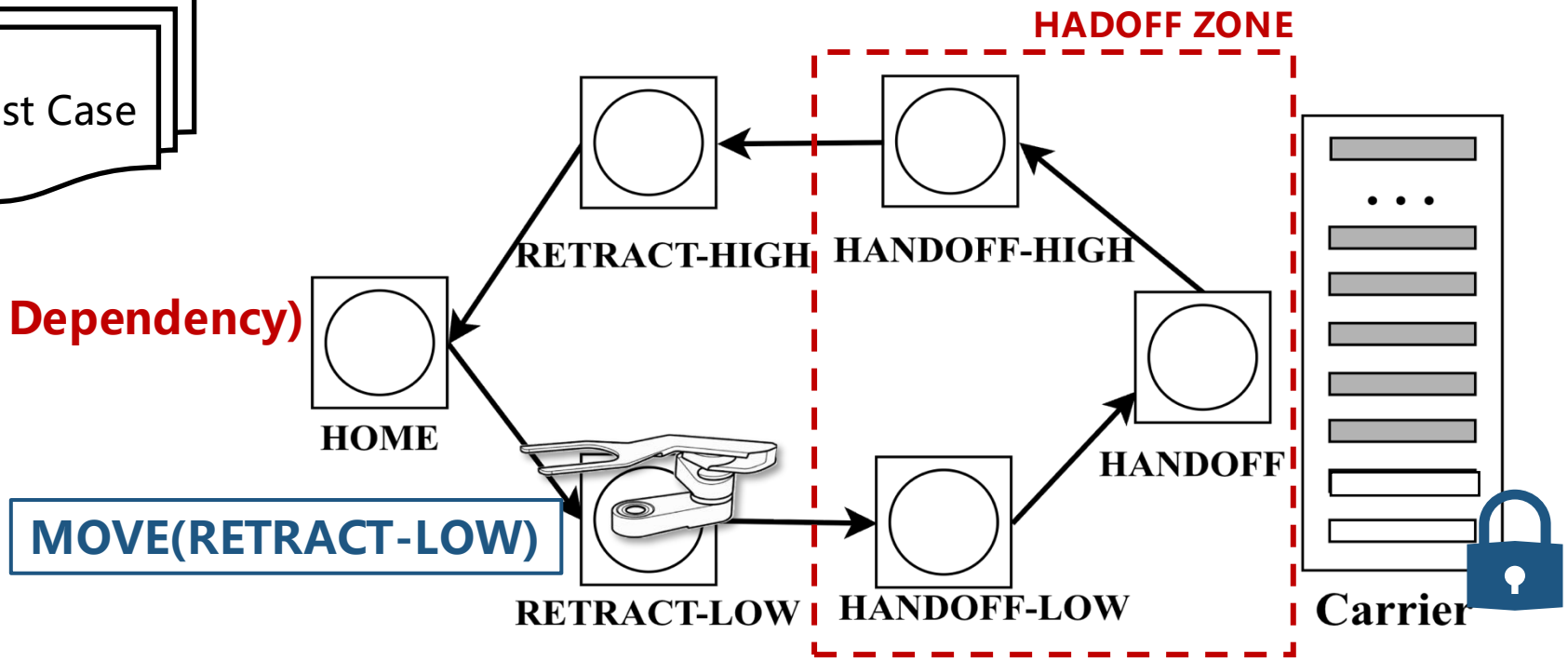
Example of Model-based Planning



Mitigate the Challenge 1 (Action Dependency)

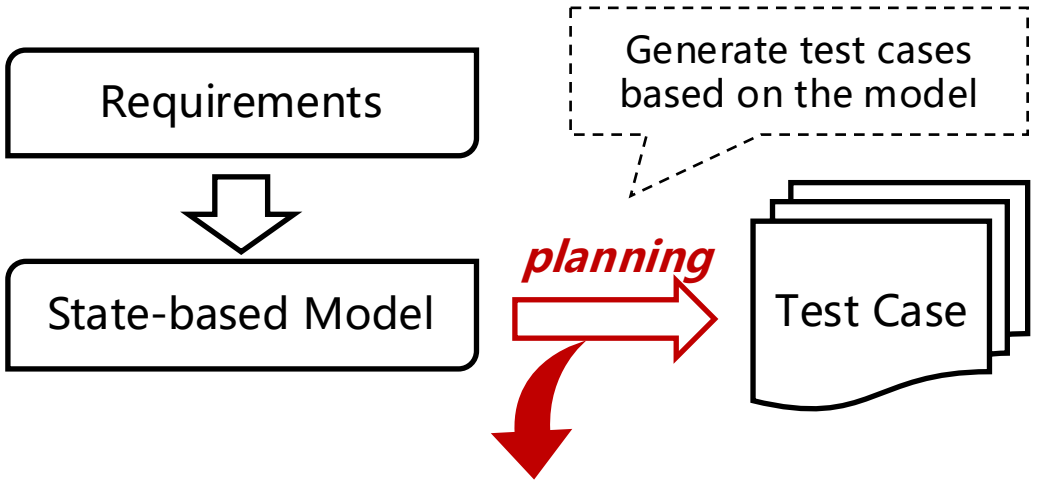
Planned Test Cases

- **MOVE(HANDOFF-HIGH)**
- **MOVE(HANDOFF)**
  PICK-WAFER
- **MOVE(HANDOFF-LOW)**
  LOCK-CARRIER



WHR is getting wafer from the carrier station by station

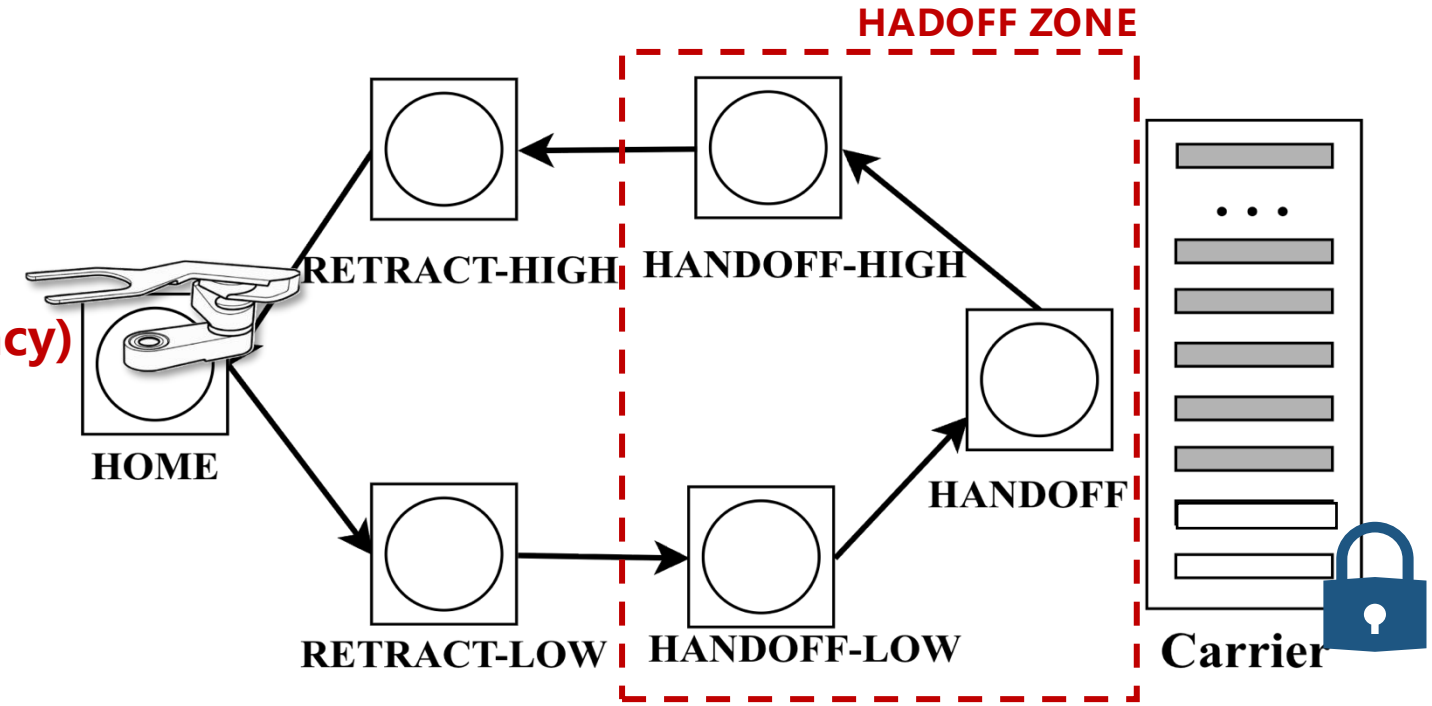
Example of Model-based Planning



Mitigate the Challenge 1 (Action Dependency)

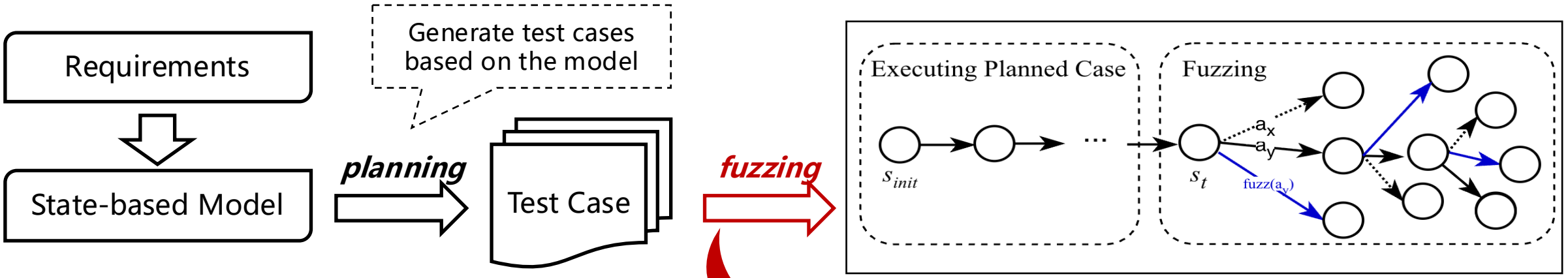
Planned Test Cases

- MOVE(HANDOFF-HIGH)
- MOVE(HANDOFF) PICK-WAFER
- MOVE(HANDOFF-LOW) LOCK-CARRIER
- MOVE(RETRACT-LOW)



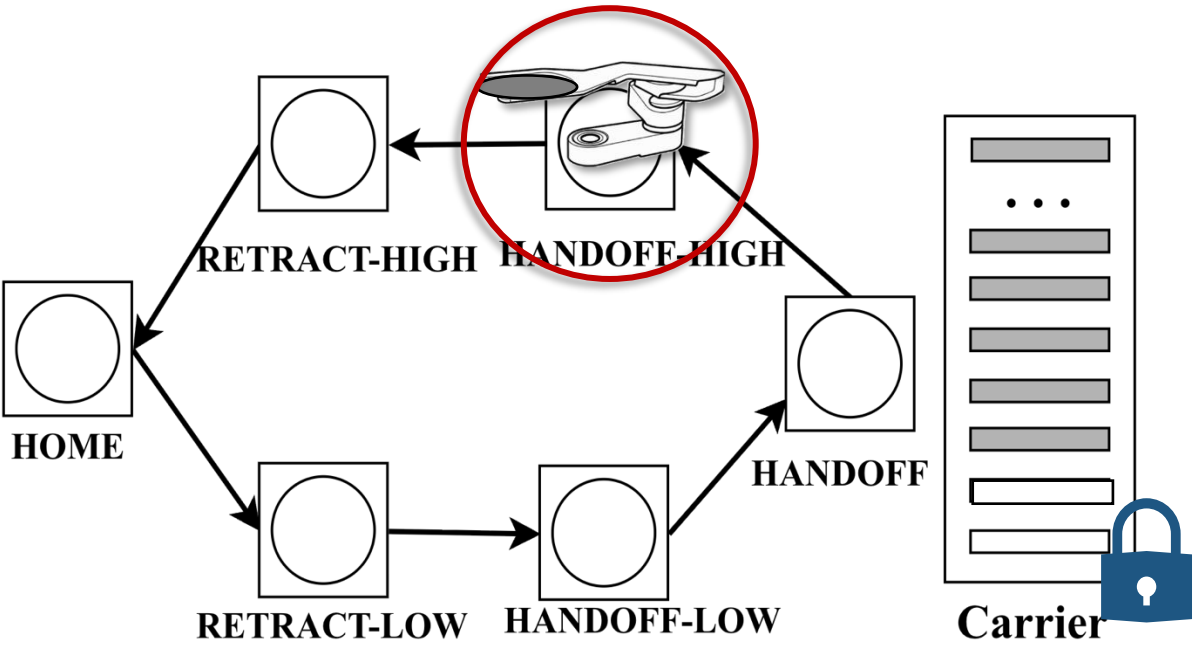
WHR is getting wafer from the carrier station by station

Approach: Model-based Fuzzing

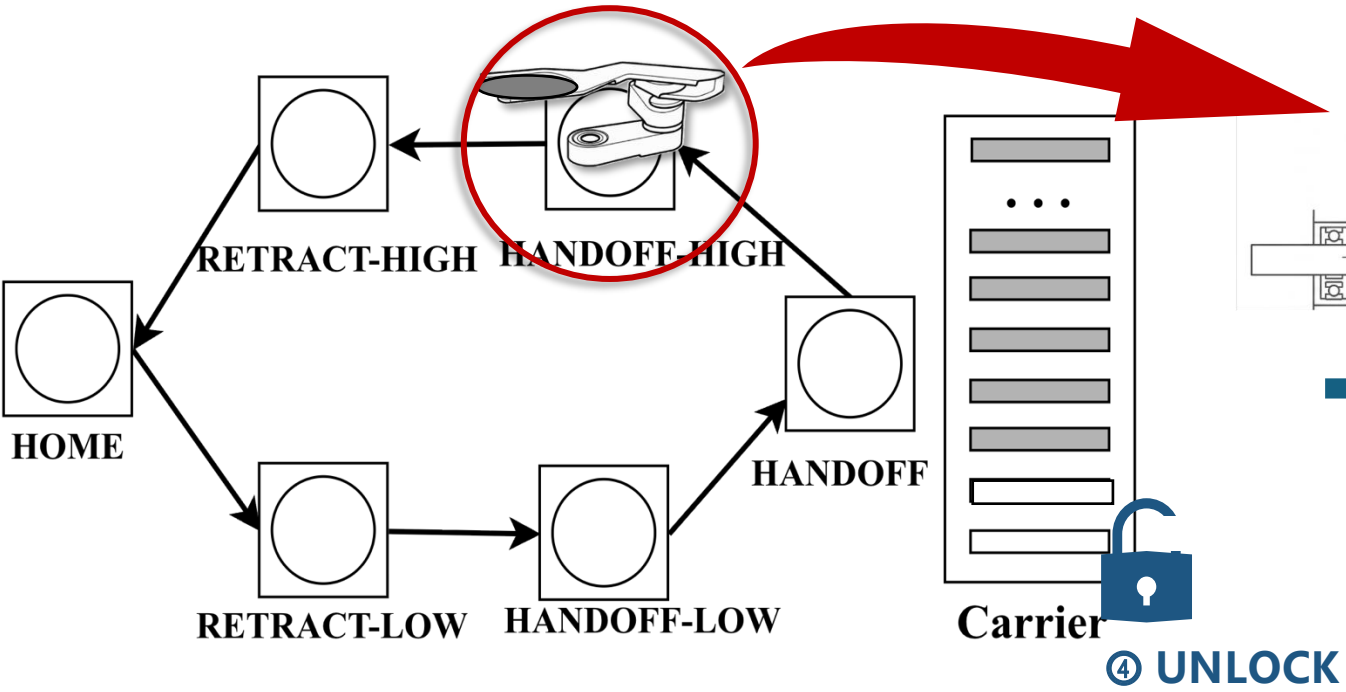


Mitigate the Challenge 2 (State Space Expsion)

Example of Model-based Fuzzing

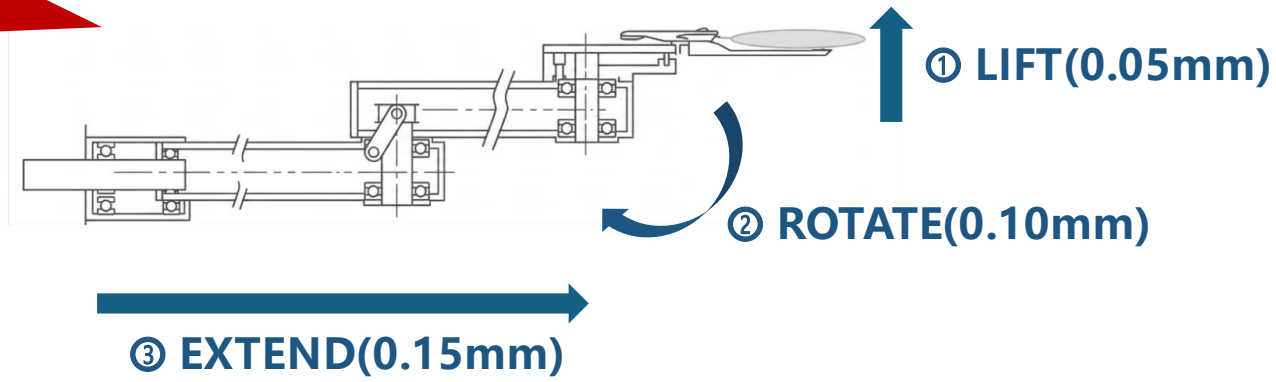


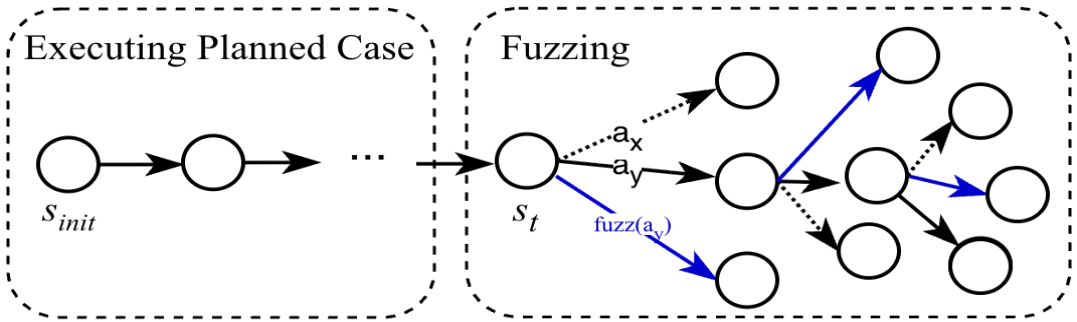
Example of Model-based Fuzzing



Fuzzing the environment

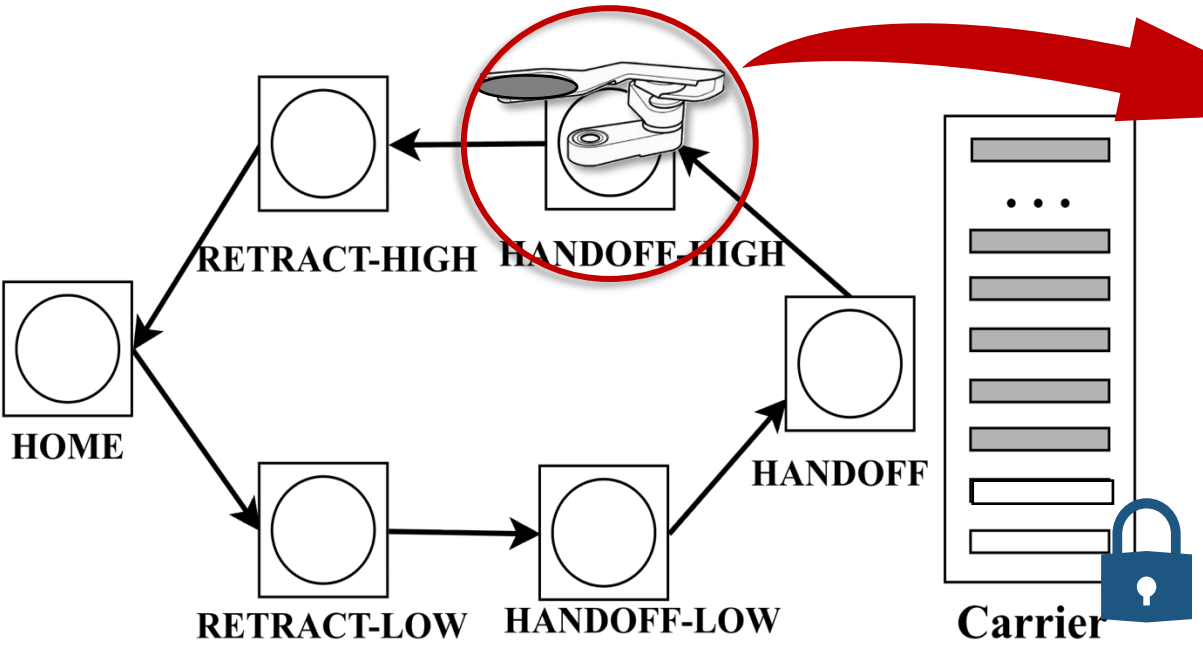
Fuzzing the action parameter



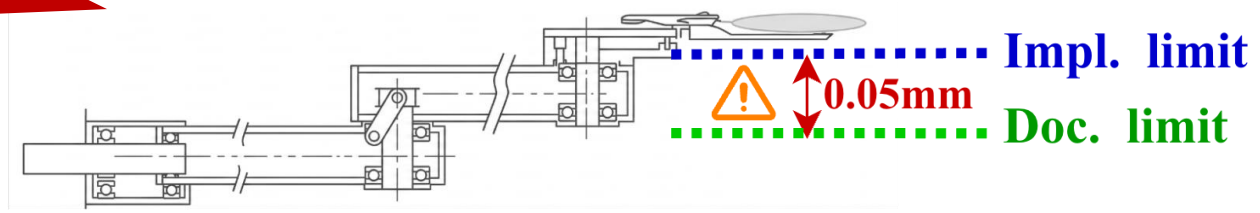


Exploring the State Space

A Real-world Bug



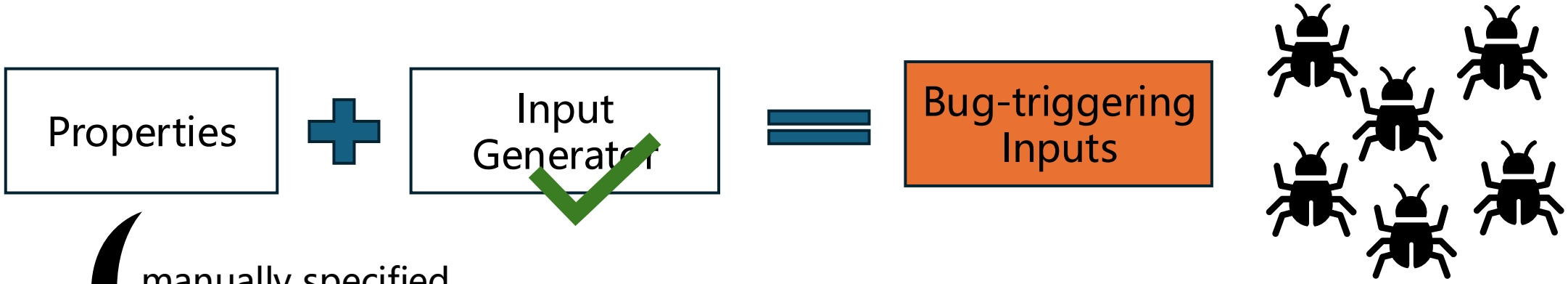
WHR is getting wafer from the carrier station by station



WHR impl. lift limit exceeds the doc. lift limit by 0.05mm at HANDOFF-HIGH with a wafer holding

may cause wafer damage or equipment collision

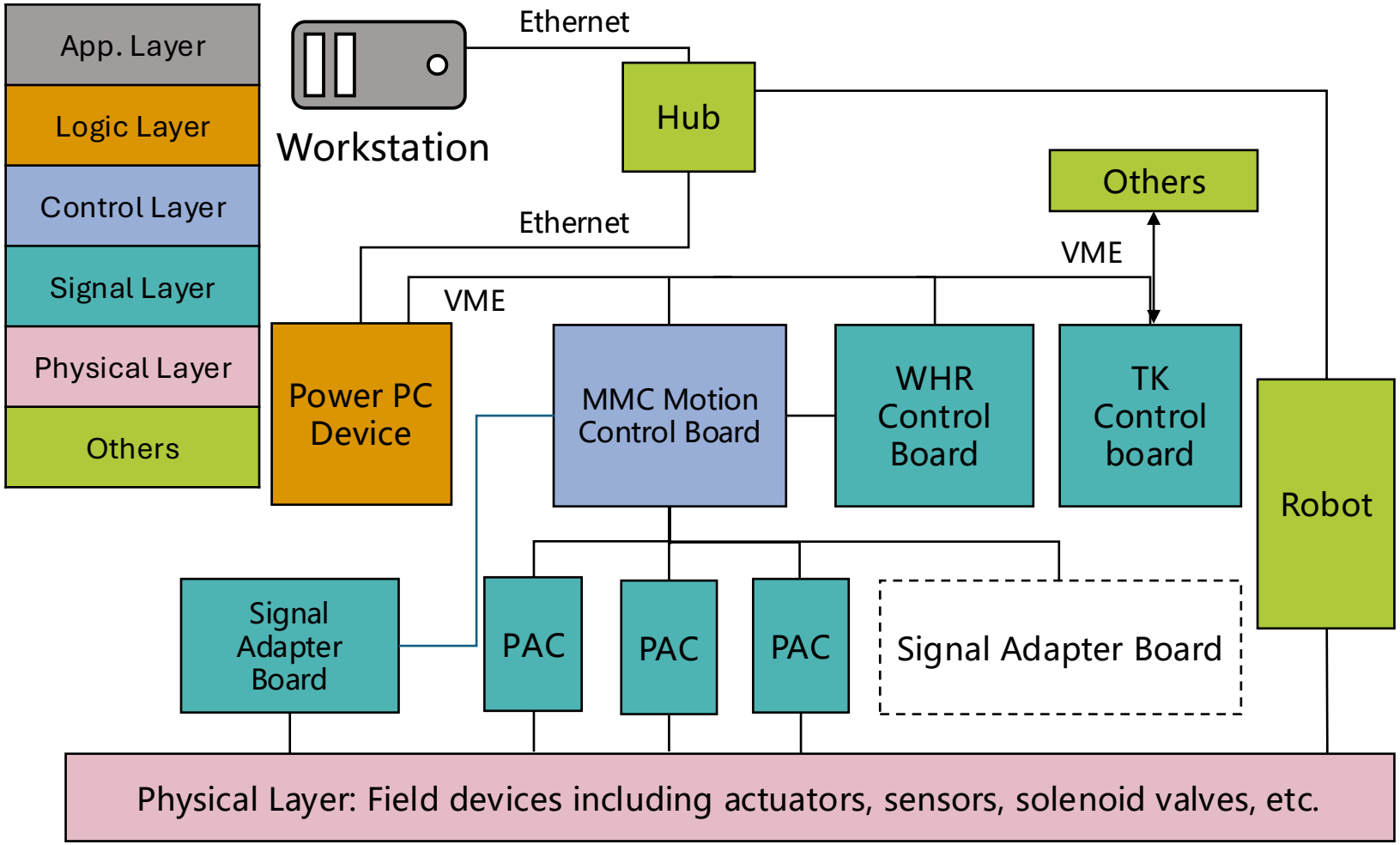
System Requirements (Properties) of WHRs



Property Name	System Requirement Description
No Crash/Hang	For any action, the WHR must run without crashing / hanging
Dual-arm Entanglement Protection	To avoid collisions, the two robot arms must not simultaneously occupy any pair of physically entangled stations
Per-Station Axis Limits	Axis movements (i.e., extension, rotation, lift) at any station strictly respect the physical bounds of that specific station.
Interaction Constraints	Environmental conditions must be satisfied whenever the WHR enters or stays in a module's handoff *zone*.
Pick/Place Consistency	A pick (place) action strictly requires an empty (occupied) gripper beforehand and results in an occupied (empty) gripper afterwards.

(For more details of WHR requirements, please refer to our paper)

Wafer Handling Adopts a Layered Architecture



Wafer Handling Adopts a Layered Architecture

Expose APIs

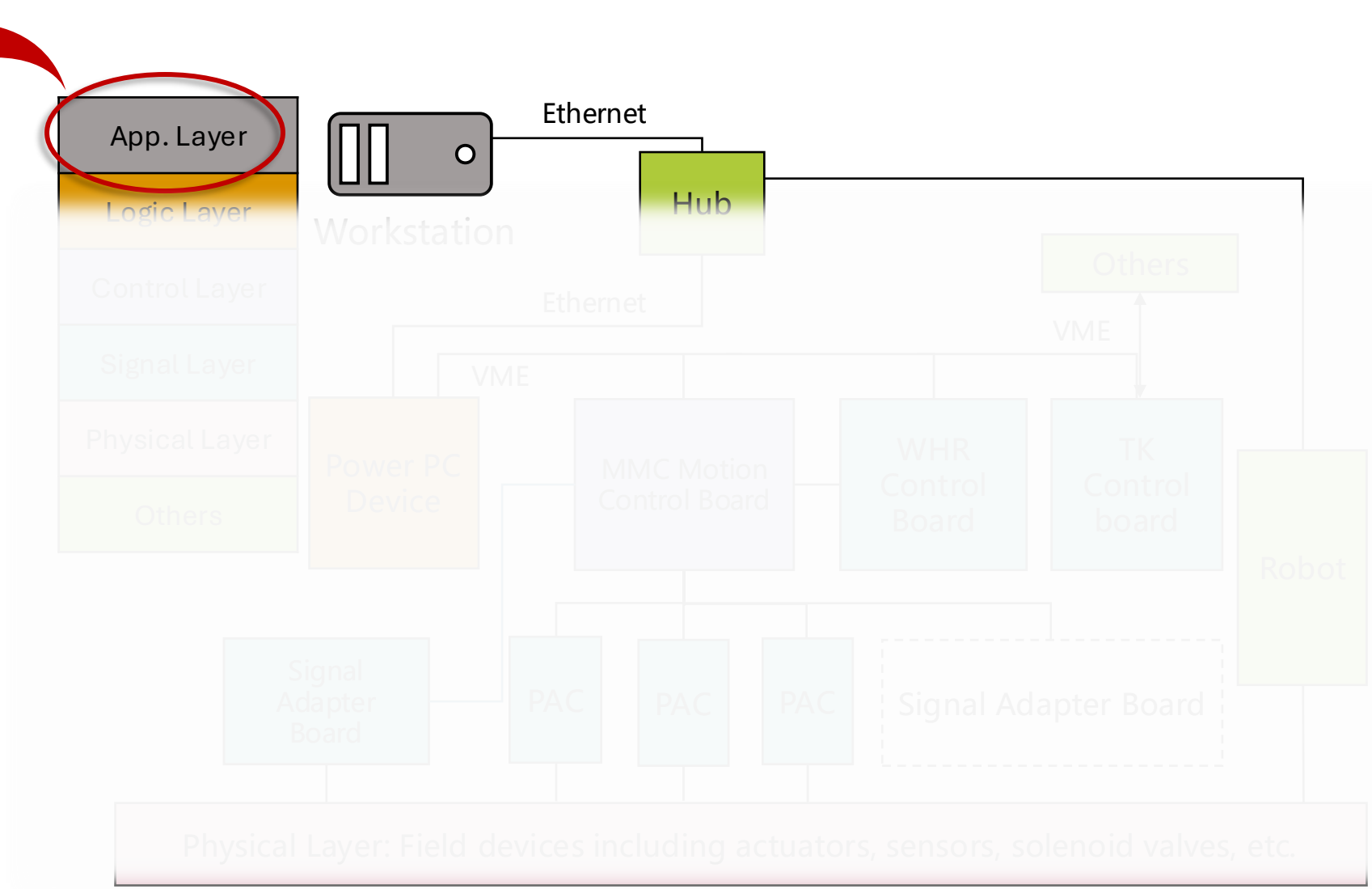
e.g.,

MOVE(STATION)

ROTATE(DISTANCE)

LIFT(DISTANCE)

… …



Evaluation Setup

- **Subject:** the latest version of a **production** WHR within industrial lithography equipment
- **Baseline**
  - **Manual Testing:** an industrial regression test suite
  - **Random Testing:** (1) random API selection & type-aware random parameter generation;
    (2) Randoop[1]: incrementally builds tests by extending successful call sequences.
- **Metrics: Bug Count + 3 Coverage Criteria** (devised by ourselves)
  - **Station Coverage:** Check if every station is visited at least once.
  - **Operation Coverage:** Execute all operations on each station.
  - **Parameter Coverage:** Exercise all parameters for valid/invalid input robustness.
- **Total Testing Time:** 10 hours

[1] C. Pacheco and M. D. Ernst. Randoop: feedback-directed random testing for Java. In OOPSLA '07

Bug Finding

- **Bug Count:** ArmGo found **21** new bugs in a production WHR
- **Categories:** Across four property types
  - ❑ no crash/hang: **1**, entanglement: **6**, axis limit: **11**, and interaction constraints: **3**).

Violated Properties	Count	Example Violation	Status
No Crash/Hang	1	Missing input validation in simulator	Fixed
Entanglement	6	Dual-arm collison protection missing	Confirmed
Axis Limits	11	Lifting limit deviation (0.05mm gap)	Fixed
Interaction Constraints	3	WHR moving within pre-aligner's area without protection checking	Fixed

(For more analysis of these bugs and lessons, please refer to our paper)

Comparison with Baselines

- **Coverage:** ArmGo outperforms all baselines across all criteria (see our paper for coverage trend)

L1: Station Coverage (Target: 100%)



L2: Operation Coverage (Industry Avg: 24%)



L3: Parameter Coverage (Target: 100%)



- **Bug Finding:** For the found 21 bugs, manual testing: **0** bug; random testing: **1** bug

(For more analysis of coverages and their trends, please refer to our paper)

Property-Based Testing of Wafer Handling Robots via Model-Based Planning and Fuzzing

- **Our Idea:** Property-based testing to validate wafer handling robots
- **Input Generation**: Model-based planning and fuzzing
- **Result: 21** previously-unknown bugs in a production WHR
- **Deployment:** Integrated into CI pipeline of our industrial partner

Scan for Paper and Artifact
(sanitized, simplified for illustration)