HD-Sec: Holistic Design of Secure Systems on Capability Hardware

DSbD
Digital Security by Design

https://hd-sec.github.io

Colin Snook, Asieh Salehi, Thai Son Hoang, Robert Thorburn, Michael Butler, Leonardo Aniello, Vladimiro Sassone
University of Southampton, UK {cfs, a.salehi-fathabadi, , t.s.hoang, robert.thorburn , m.j.butler, l.aniello, vsassone}@soton.ac.uk

University of Southampton
UKRI Engineering and Physical Sciences Research Council

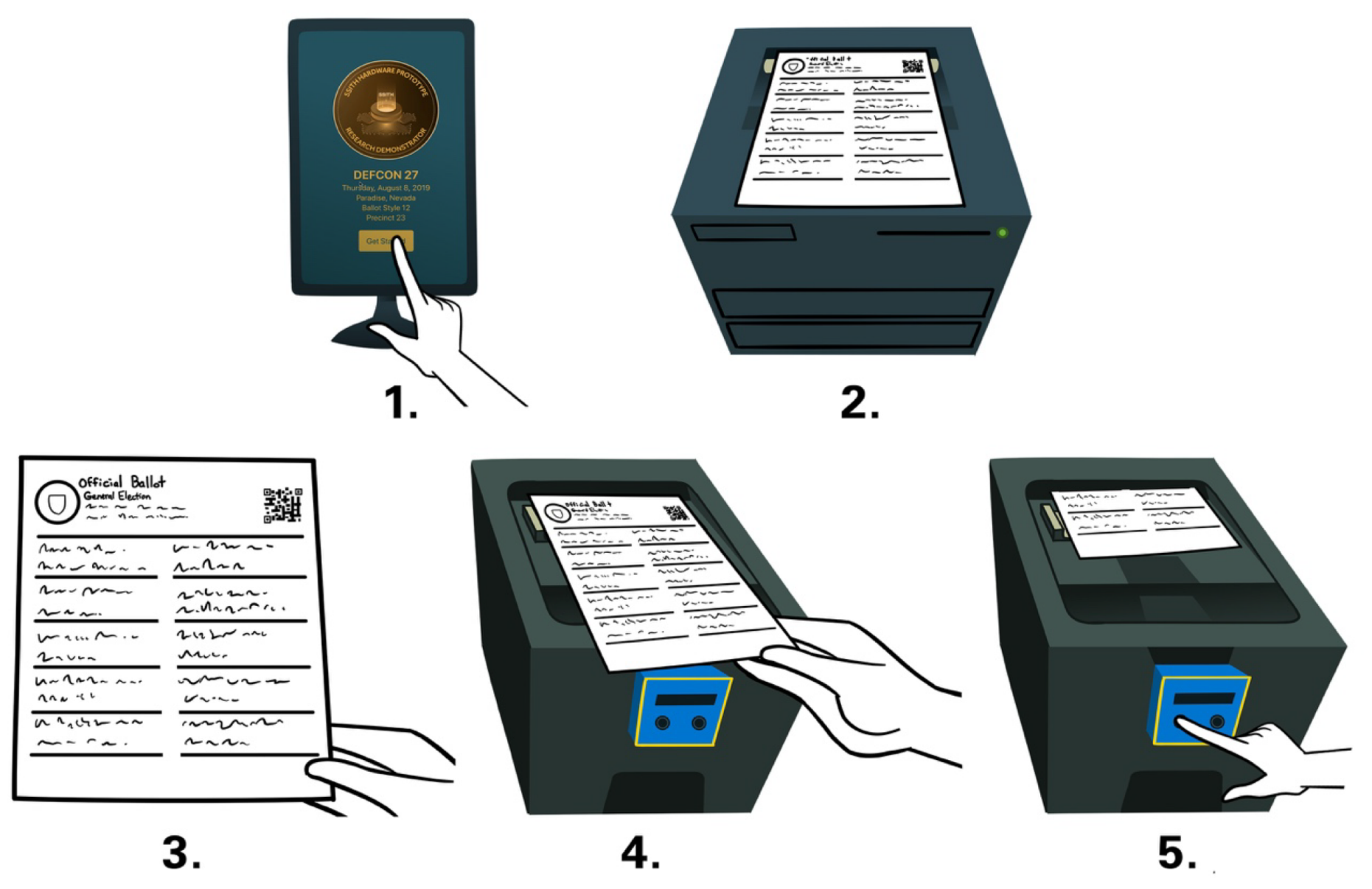
Project Overview

Transformation of security-critical software development

- From an expensive iterative test-and-fix approach
  - To a correctness-by-construction (CxC) approach
- The design of software from requirements to implementation
  - Formal modelling, Reusable formal abstractions
  - Verification
  - Model transformation
  - CxC tools and running on capability hardware

Case Study: Smart Ballot Box[1] (SBB)

- **Availability:** the voter should not be prevented from casting a ballot.
- **Confidentiality :** the voter's choices should be secret
- **Integrity:** the system should only accept valid ballots and reject invalid ballots.



Designing Error handling for CHERI exceptions

- Error handling allows runtime management of hardware-detected memory attacks, security solutions can be designed into the system.
- Exception handling needs to be context and application specific:
  - UML-B modelling to develop application-specific error handling behaviour for critical component

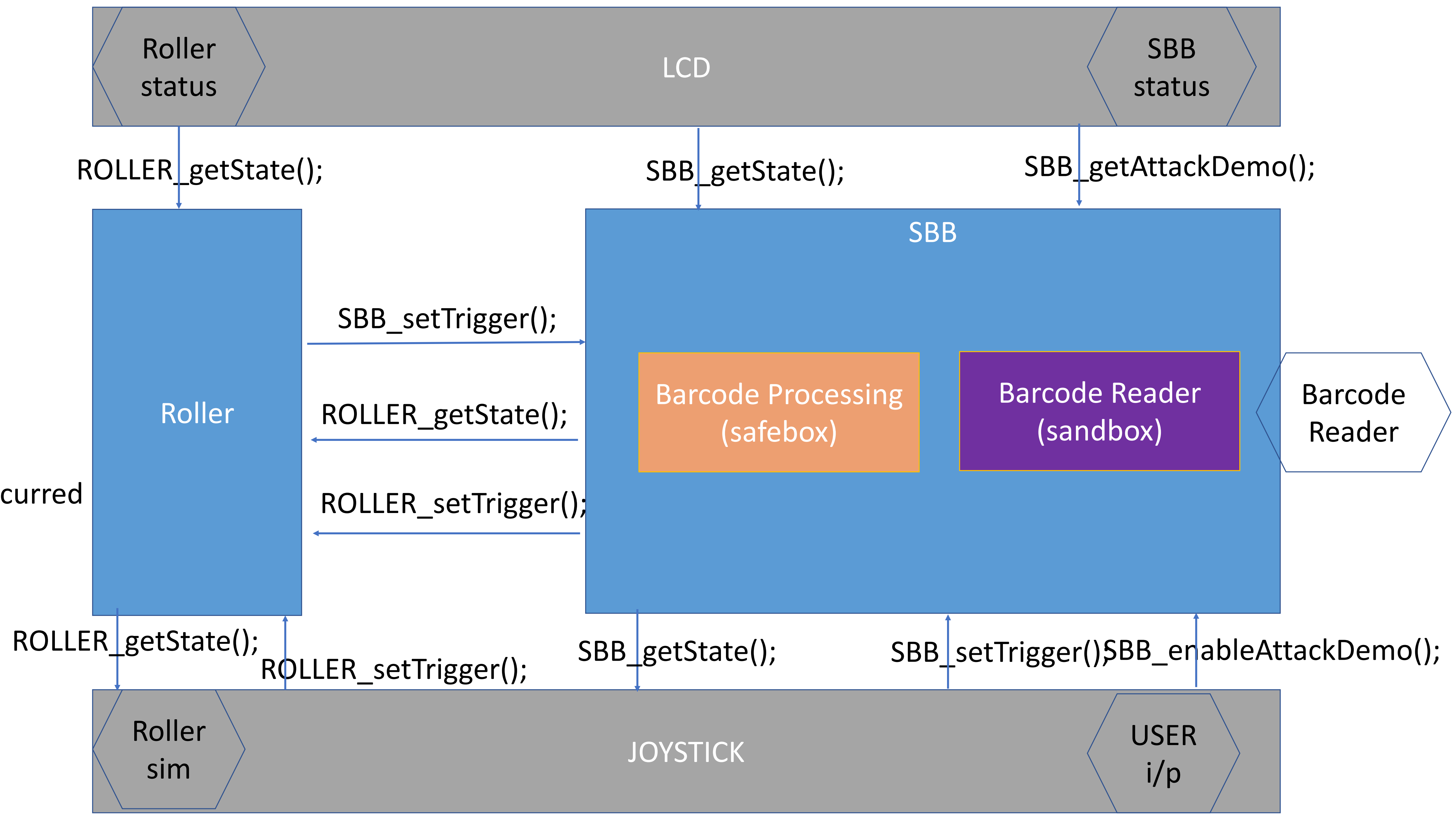
SBB Sonata - Compartment structure

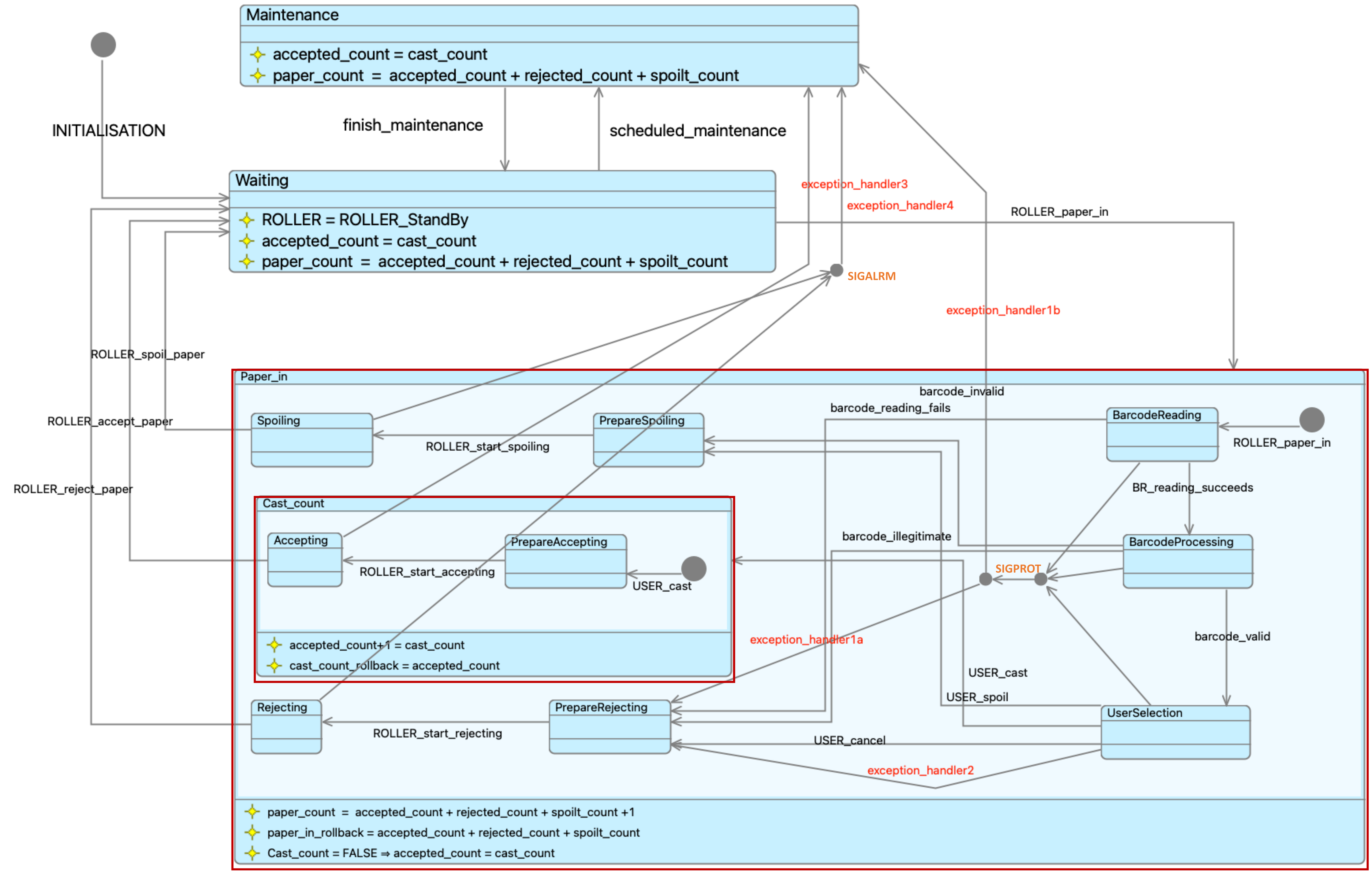
- CHERI compartments provide support for
- structuring/containment of error-handling
- data encapsulation
- Two Top level Thread compartments
- Roller and SBB
- Two Library compartments
- Barcode Reader and Barcode Processing

SBB formal model - Implementation for Sonata board

- State encoding of implementation enables rich error handling
- Error handler uses state to determine where the exception occurred
- Error handler uses state to initiate a recovery state

- Statemachines implemented as *switch (state)*
- Transitions implemented as *functions*
- Exception transitions are implemented In the Cheri *compartment_error_handler*





References

[1] Galois and Free & Fair. The BESSPIN Voting System, 2019.
[2] Designing exception handling using Event-B, In ABZ 2024: Rigorous State-Based Methods.
[3] Analysing the safety implications of security risks in cyber-physical systems, In The Practice of Formal Methods, Springer LNCS 14781, 2024.
[4] An Event-B Formal Model for Access Control and Resource Management of Serverless Apps, In ABZ 2024: Rigorous State-Based Methods.
[5] Systematic hierarchical analysis of requirements for critical systems In Innovations in Systems and Software Engineering (A NASA Journal), 2024.
[6] CuneiForm Method for Assuring the Safety of ML-Based Computer Vision Development Datasets, IEEE 32nd International Requirements Engineering Conference Workshops, 2024.