National Cyber
Security Centre

Methodological Findings from Applying STPA in Cyber Security Case Studies

Dr Anna G. – Sociotechnical Security Researcher

UK National Cyber Security Centre

Methodological Findings from Applying STPA in Cyber Security Case Studies

- Intro to the role of the UK National Cyber Security Centre (NCSC)

- Our Work with STAMP and STPA

- Methodological Findings:
  - Type B Scenario Generation
  - Documentation of additional information such as subsystem states and conditions

UK National Cyber Security Centre

Vision:
To make the UK the safest place to live and work online

Act as a bridge between industry, government and academia

Unified source of advice, guidance and support on cyber security

Sociotechnical Security Group

Cyber security research in practice

Sociotechnical lens on cyber security problems

Multidisciplinary

Interactions between people, technology, organisations and processes

Our Work with STAMP and STPA

Risk Frameworks – Core Research Questions:

Do we have the right mix of
tools / techniques / frameworks
for the cyber security problems of today
and in the future?

If not,
what do we need to ensure
our cyber security risk toolbox is fit
for the cyber security problems of today
and in the future?

Systems theoretic approaches
to cyber security risk,
and STAMP in particular,
should be part of our cyber security
risk toolbox.

Our Work with STAMP and STPA

Exploring applicability to a variety of different use cases:

Traditional cyber security scenarios

- Enterprise IT infrastructure

Joint safety and cyber security contexts

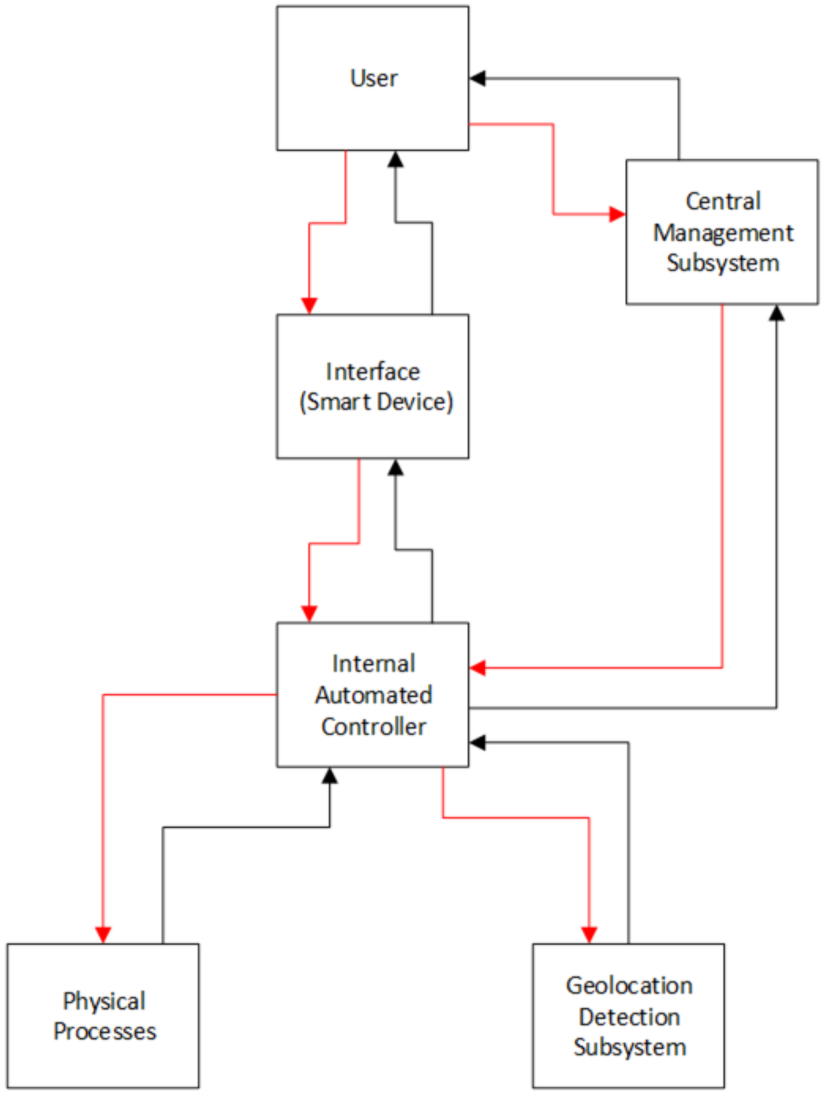
- Automated / connected products
- Industrial control systems
- Critical national infrastructure

Number of case studies working with UK stakeholders involving systems in design and in operations

Illustrative Example – Drone

Key Points

- Case study involving an automated product in design
- User interface such as a smart device
- Safety and security concerns
- Completed several STPA iterations
- Increasingly detailed and complex HCS



Methodological Findings: Type B Scenario Generation

STPA Step 4: Identify Loss Scenarios and Requirements

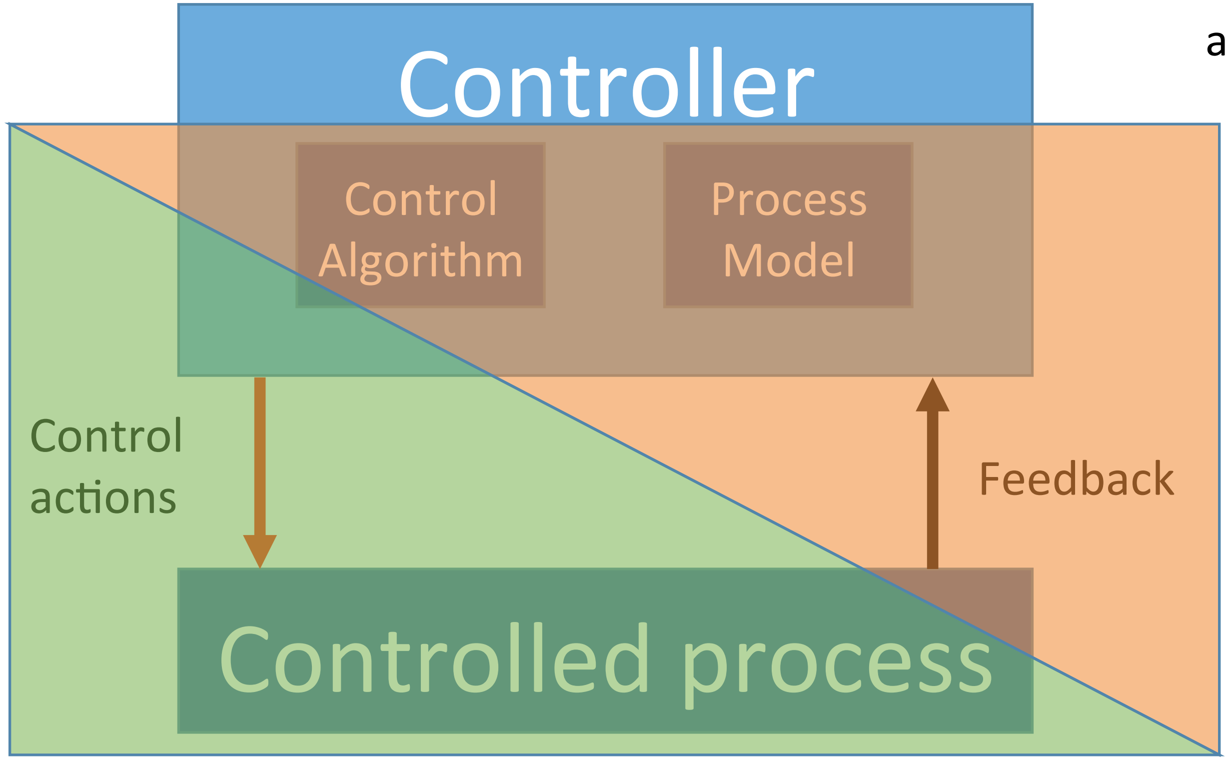
Our original method applied in case studies
- Take each individual UCA identified in Step 3
- Apply Type A scenario thinking to the UCA
- Apply Type B scenario thinking to the UCA

Too limited
- Type B scenarios linked directly to hazard
- Can apply Type B to control actions

But not want to lose relationship between UCAs and both types of scenarios

Type A

Why would an Unsafe Control Action occur?



Type B

Why would control actions be improperly executed or not executed, leading to hazard?

Type B Scenario Generation
How to generate the broadest range of Type B scenarios to inform subsequent requirements?

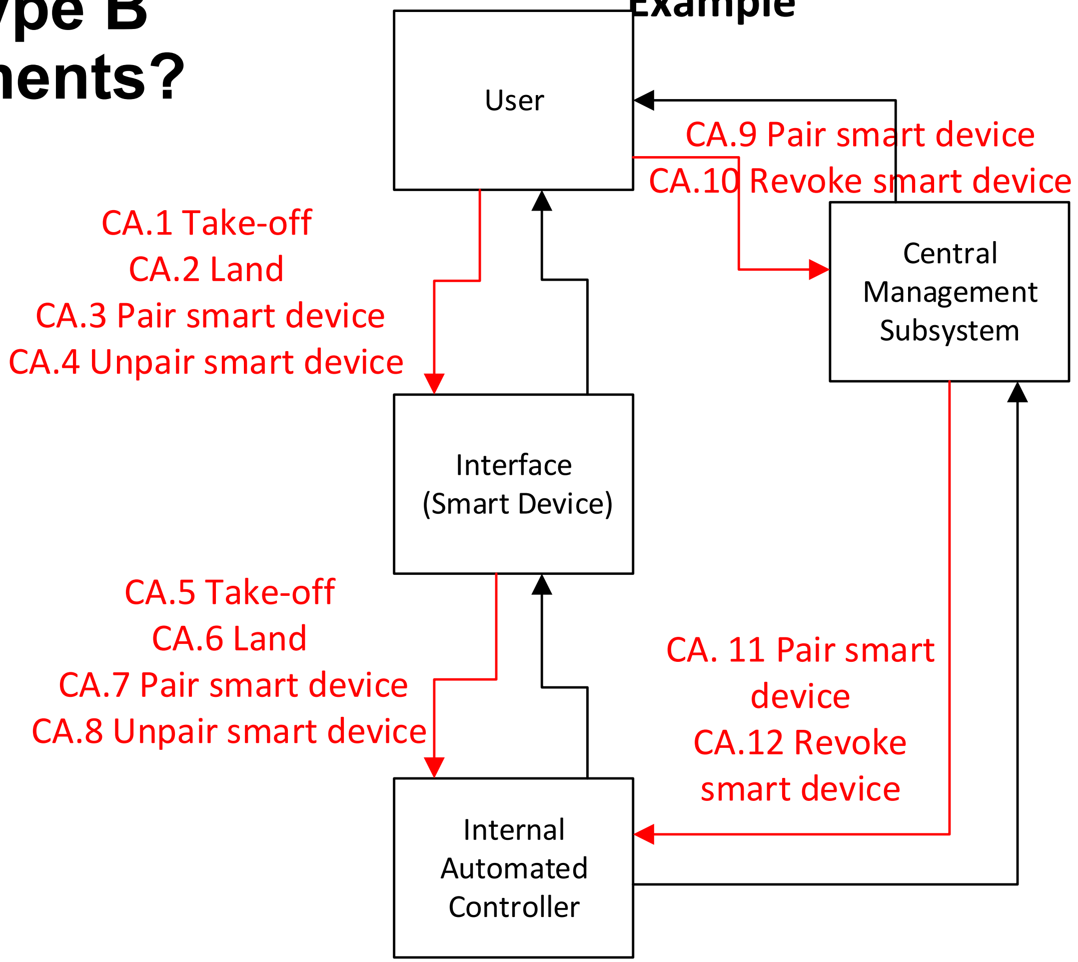
Adjusted methodology applied in case studies:

- Take each individual UCA identified in Step 3
- Apply Type A scenario thinking to the UCA
- Apply Type B scenario thinking to the UCA

- Apply Type B scenario thinking to the control action as a whole

- Consider requirements generated from both Type A and B scenarios applied to the individual UCAs when generating requirements to mitigate Type B scenarios from corresponding Control Action

Illustrative Drone Example

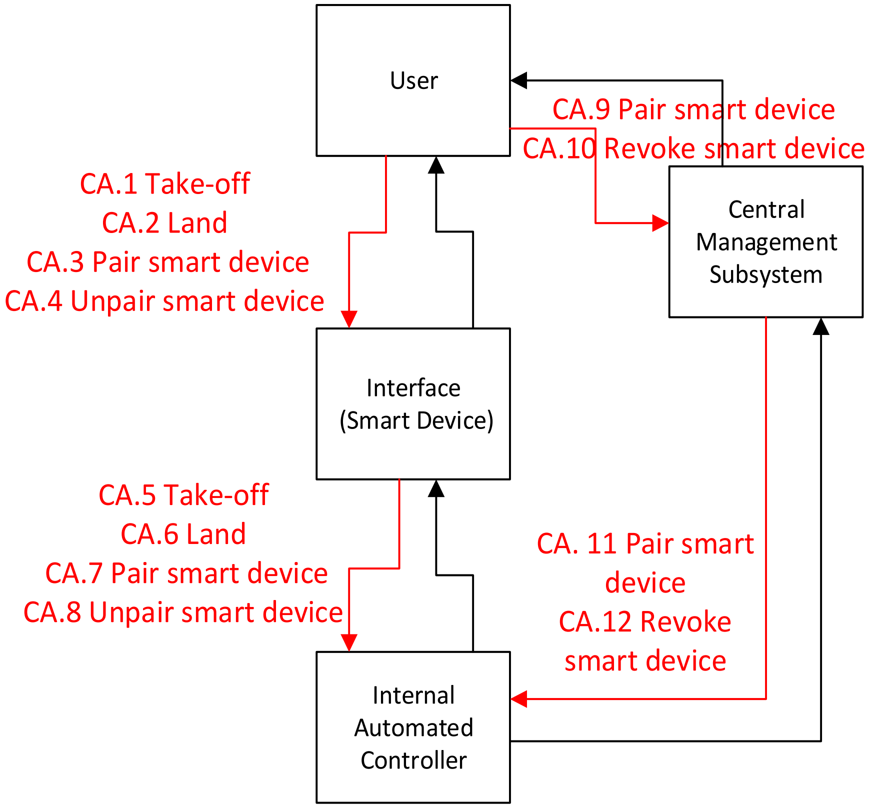


Interplay between Type A and Type B Scenarios and Requirements

Type B Scenario analysis applied to CA.5 ‘Take-off’ and CA.6 ‘Land’

Serial	From	To	Action	Type B Scenario Description	Hazard	Additional Requirements
CA.5	Interface (Smart Device)	Internal Automated Controller	Take-off	These scenarios refer to a situation in which the commands are not actioned. This could occur due to a failure in the control path, either by a malicious actor jamming the connection, or by a technical failure. There is also a possibility that legitimate commands from the user would be countermanded in the control path by a spoofed smart device. These risks have already been mitigated by R3.5 and R.3.9.	H.02, H. 03	None – exposure to hazard mitigated by existing requirements.
CA.6	Interface (Smart Device)	Internal Automated Controller	Land			

Illustrative Drone Example



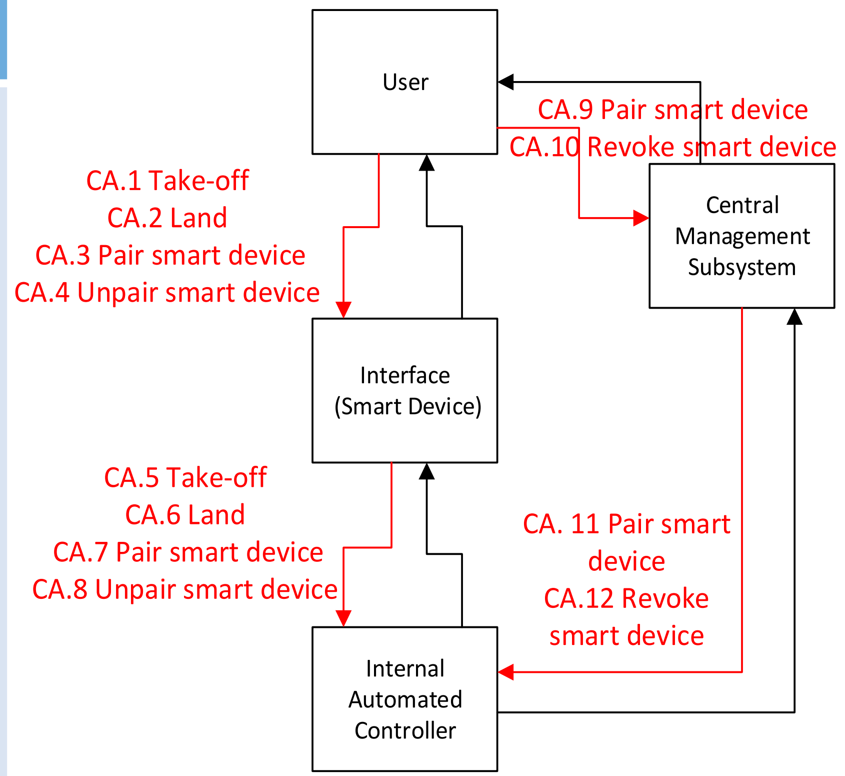
National Cyber Security Centre

Interplay between Type A and Type B Scenarios and Requirements

Type B Scenario analysis applied to CA.12 Revoke smart device

Serial	From	To	Action	Type B Scenario Description	Hazard	Additional Requirements
CA.12	Central Manage-ment Subsystem	Internal Automated Controller	Revoke smart device	In this scenario the CA 'Revoke smart device' is not received or actioned by the Internal Automated Controller. This could allow control actions from a stolen or spoofed smart device to continue to exert control over the drone. Currently commands from the smart device and the central management system could be received contemporaneously and those from the smart device could be actioned, overriding those from the central management system. Mitigation would be to privilege the commands from the central management subsystem over other controllers.	H.01, H.05	R.3.28 There should be a mechanism to ensure that commands from the Central Management System are given precedence over commands from other controllers.

Illustrative Drone Example



Interplay between Type A and Type B Scenarios and Requirements

What did this approach give us?

- Broad basis for generating both types of scenarios and corresponding requirements

- Utility in practice of considering the potential exposure to hazard from different directions

- Found new scenarios and additional requirements

- Interplay between scenarios and requirements generated from individual UCAs and the control action the UCA is derived from

Requirement	Derived from:	Connection to Hazard
R3.5	UCA3.2 - Type A CA.5 - Type B CA.6 - Type B	H.02 H.03
R3.9	UCA3.2 - Type A CA.5 - Type B CA.6 - Type B	H.02 H.03
R3.28	CA.12 - Type B	H.01 H.05
……	…..	……

- Traceability of requirements to multiple scenarios and exposure to hazard

- Added weight to necessity of requirements when communicating findings

Methodological Findings: Documentation of Subsystem States / Conditions

- Case Study Example Key Points:

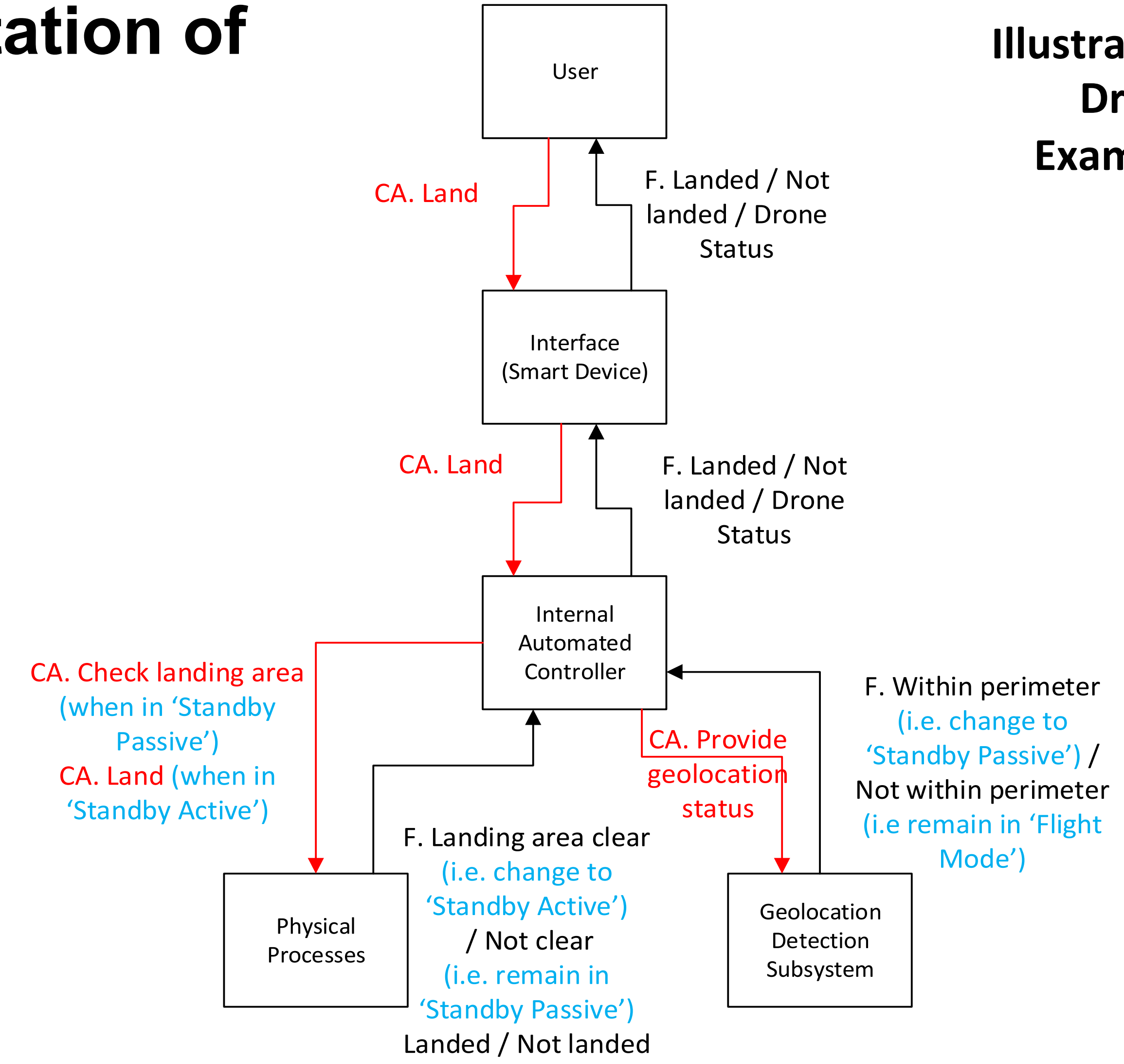
- Automated product in design

- Safety and security concerns

- Geo-fenced perimeter for landing

- Importance of:
  - Sequencing of available control actions
  - Moving between states of 'Disabled', 'Flight Mode', 'Standby Passive' and 'Standby Active'

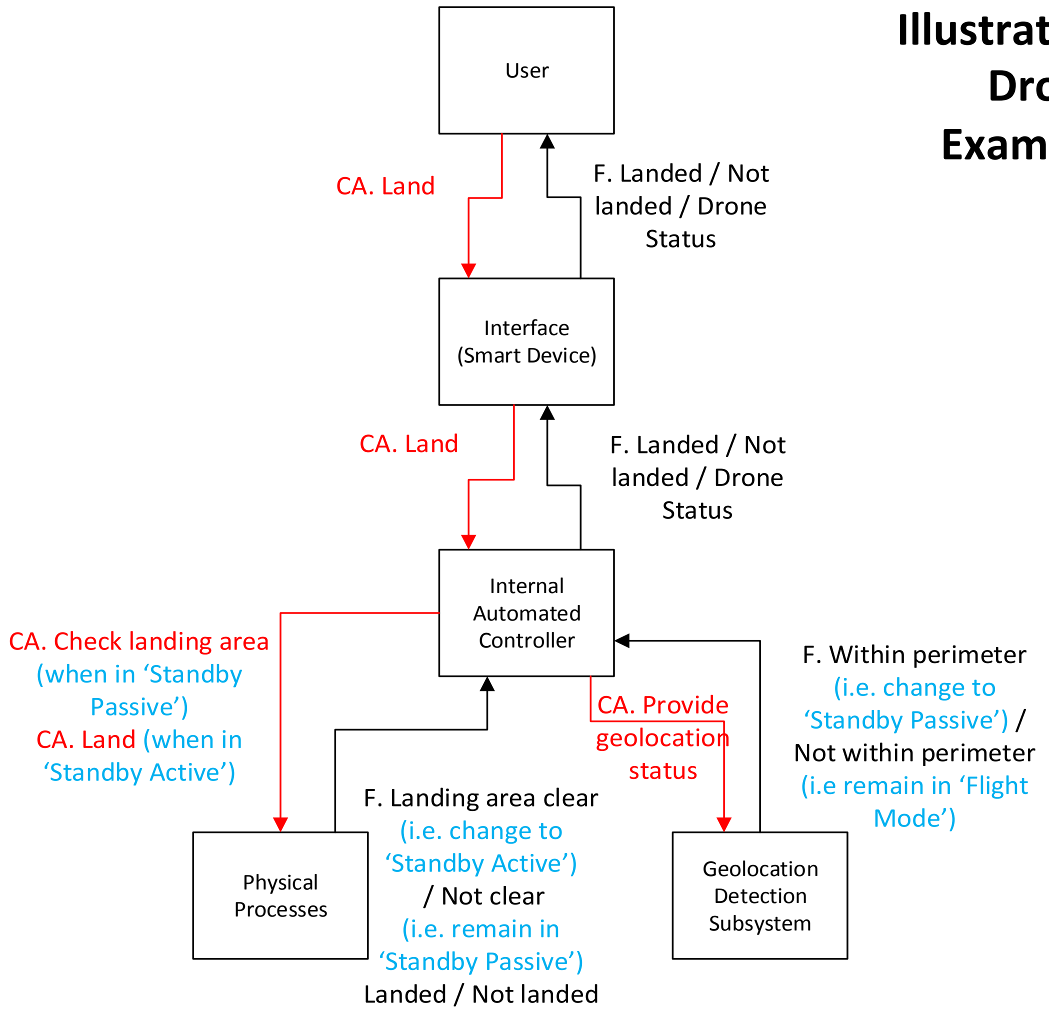
Illustrative Drone Example



Documentation of Subsystem States / Conditions

Illustrative Drone Example

From	To	Control Action	When this condition is true:	Feedback	Change to status?
User	Interface	Land	Standby Passive or Standby Active	Landed Not Landed Drone Status	N/A
Automated Internal Controller	Geolocation Detection Subsystem	Provide geolocation status	All states	Within perimeter Not within perimeter	Standby Passive No change
Automated Internal Controller	Physical Processes	Check landing area	Standby Passive	Landing area clear Not clear	Standby Active No change
Automated Internal controller	Physical Processes	Land	Standby Active	Landed Not landed	N/A



Documentation of Subsystem States / Conditions

From	To	Control Action	When this condition is true:	Feedback	Change to status?
User	Interface	Land	Standby Passive or Standby Active	Landed Not Landed Drone Status	N/A
Automated Internal Controller	Geolocation Detection Subsystem	Provide geolocation status	All states	Within perimeter Not within perimeter	Standby Passive No change
Automated Internal Controller	Physical Processes	Check landing area	Standby Passive	Landing area clear Not clear	Standby Active No change
Automated Internal controller	Physical Processes	Land	Standby Active	Landed Not landed	N/A

Helps define what options are available under what conditions to form part of Control Algorithm of a Controller

Helps define what feedback a Controller needs for its Process Model and what it needs to know about the state of the system

Documentation of Subsystem States / Conditions

Additional information to be recorded:
- Subsystem states
- Conditions that must be true for transitions between such states
- Subsequent changes to status dependent on what feedback is received

May help analyst to spot:
- Missing subsystem states
- Missing conditions necessary for transitions
- Sequencing errors leading to hazard

May help analyst to generate:
- UCAs
- Loss scenarios
- Requirements to mitigate exposure to hazard

From	To	Control Action	When this condition is true:	Feedback	Change to status?
User	Interface	Land	Standby Passive or Standby Active	Landed Not Landed Drone Status	N/A
Automated Internal Controller	Geolocation Detection Subsystem	Provide geolocation status	All states	Within perimeter Not within perimeter	Standby Passive No change
Automated Internal Controller	Physical Processes	Check landing area	Standby Passive	Landing area clear Not clear	Standby Active No change
Automated Internal controller	Physical Processes	Land	Standby Active	Landed Not landed	N/A

Dependent on system under analysis
- Level of complexity / detail of the HCS
- Number of subsystem states / conditions

Our Next Steps

- Continue to deepen our understanding of STAMP (STPA and CAST) in relation to cyber security
- Provide advice and guidance as applicable across our broad remit
- Expand the systems theoretic approaches available in our cyber security risk toolbox

Questions?

Contact: anna.g@ncsc.gov.uk