Activity diagram legend

● Start

◉ End

◇ Condition

[...] Guard

▭ Activity/state

▭ o–o Composite activity/state

▱ Send signal

▷ receive signal

▭ comment

Create aircraft processes

resource manager o–o

input.txt
~
~
~

parser o-o

Radar

PSR o–o → SSR o–o

PSR

scan o-o

Send illuminated objects [Continue] [terminate]

---

scan

emitPulse → receiveEchoes → ⬦ [True] → determineDistances

[False]
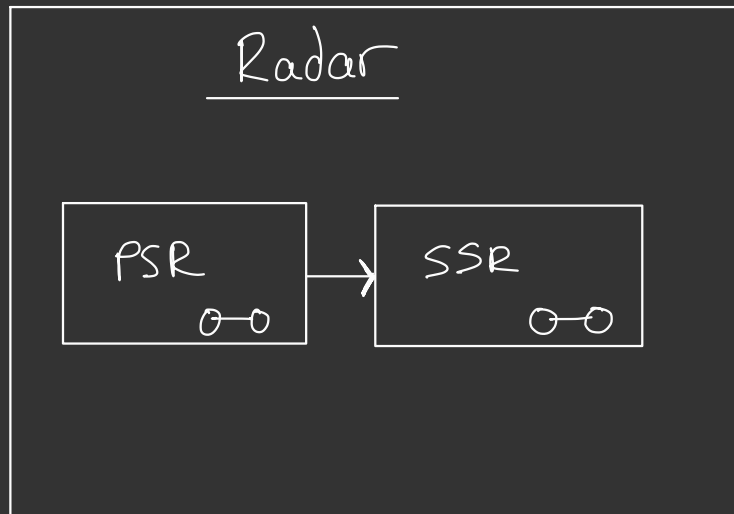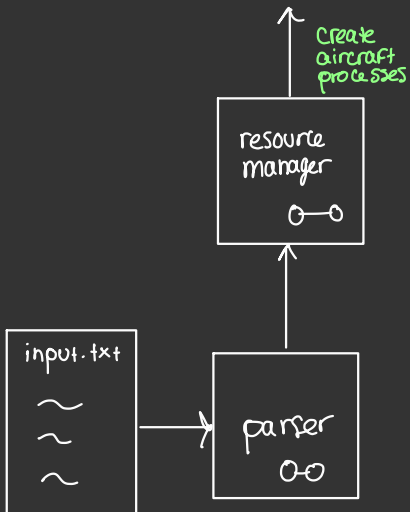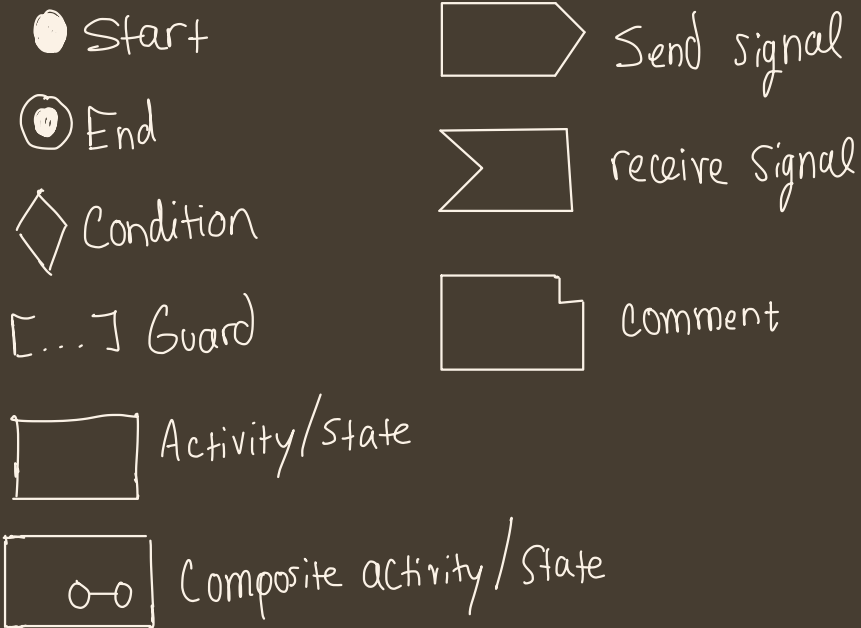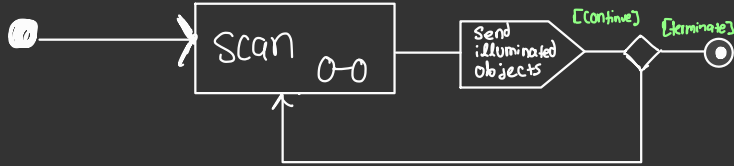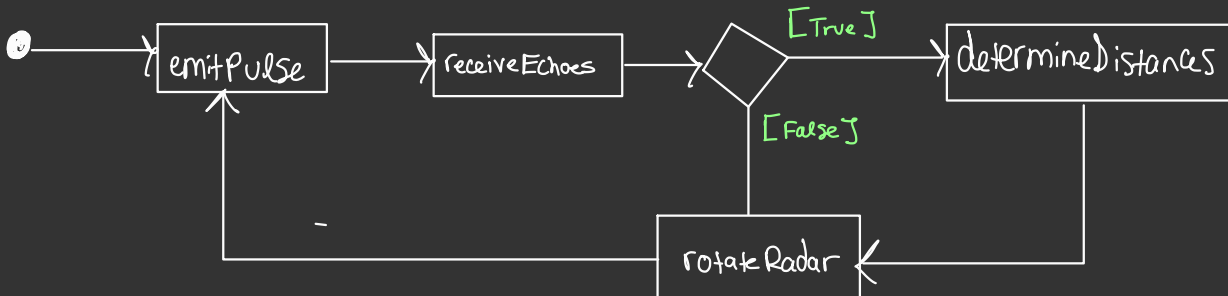
rotateRadar

PSR
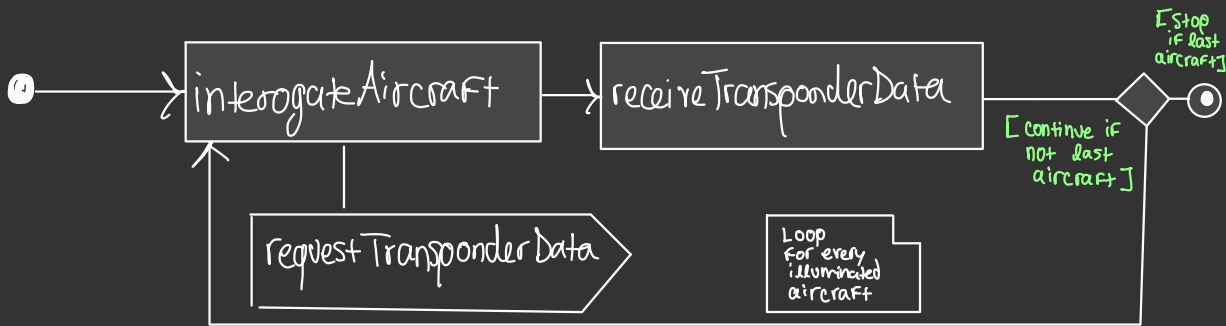
- range : int
- reference_angle : int
- scans_per_minute : int
- pulses_per_degree: int
- current_angle : int
- pulse_speed : int

+ PSR (...)
+ scan ( ) : void
- emitPulse ( ) : void
- rotateRadar ( ) : void
- receiveEchoes (echoes : int [ ]) : bool
- determineDistances (echoes: int [ ]) : Aircraft [ ]
- sendAircraftPositionsToCS (aircraft : Aircraft [ ]) : void
- sendAircraftPositionsToSSR (aircraft : Aircraft [ ]) : void

SSR

```
>receive          interrogate          Send
 illuminated          o-o              transponder
 aircraft                              data
```

interrogate

> interogate Aircraft → receive Transponder Data

request Transponder Data

[stop if last aircraft]

[continue if not last aircraft]

Loop for every illuminated aircraft

SSR

```
+ SSR( )
+ interrogate (illuminatedObjects : Aircraft [ ]) : Aircraft [ ]
+ interrogate Aircraft ( Aircraft  targetAircraft) : void
+ receive Transponder Data (Aircraft  targetAircraft ) : void
+ send Transponder Data (transponder Data : Aircraft [ ]) : Aircraft [ ]
```