SIEMENS

Railway Control with a Raspberry Pi

University of Strathclyde - Glasgow

22nd April 2017

Who are Siemens?

- Siemens is the largest manufacturing and electronics company in Europe
  - Siemens' UK operations were founded 174 years ago by Sir William Siemens, and we currently employ 14,000 people in the UK.

SIEMENS

Ingenuity for life

SIEMENS

Who are Siemens?

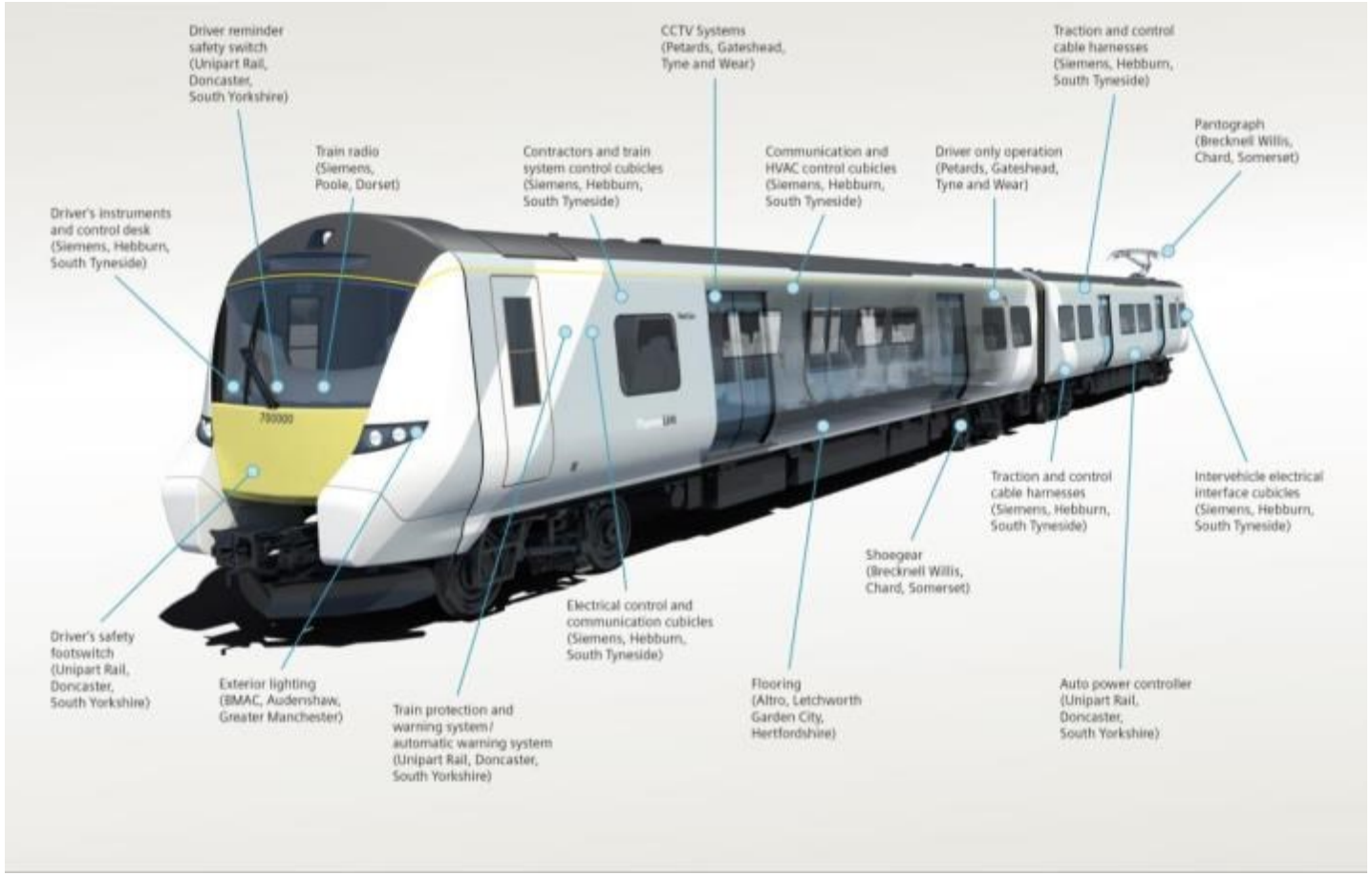
- Our goal is to improve quality of life through electrification, automation, and digitalisation.
  - e.g. State-of-the-art megnetic imaging technology;
    £160 million investment in wind turbine assembly;
    Removing 1m cars worth of CO2 through efficient drive technologies

SIEMENS

Ingenuity for life

Who are Siemens?

- Siemens Mobility develop the rolling stock and signalling control systems that allow safe and sustainable growth of our public transport network.



What do we do?

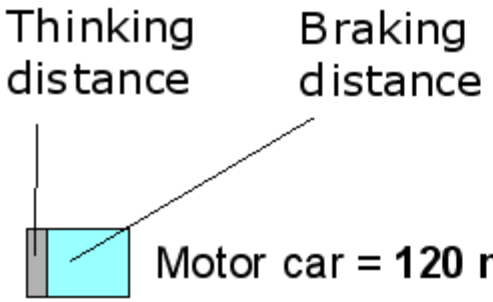
What do we do?


1 Hainault via Newbury Park
Calling at: Hainault.
1 min
4 Epping
6 mins
16 59:10
WITH MORE PROPERTIES
AN ANYWHERE ELSE
Way out
Northern line

What do we do?

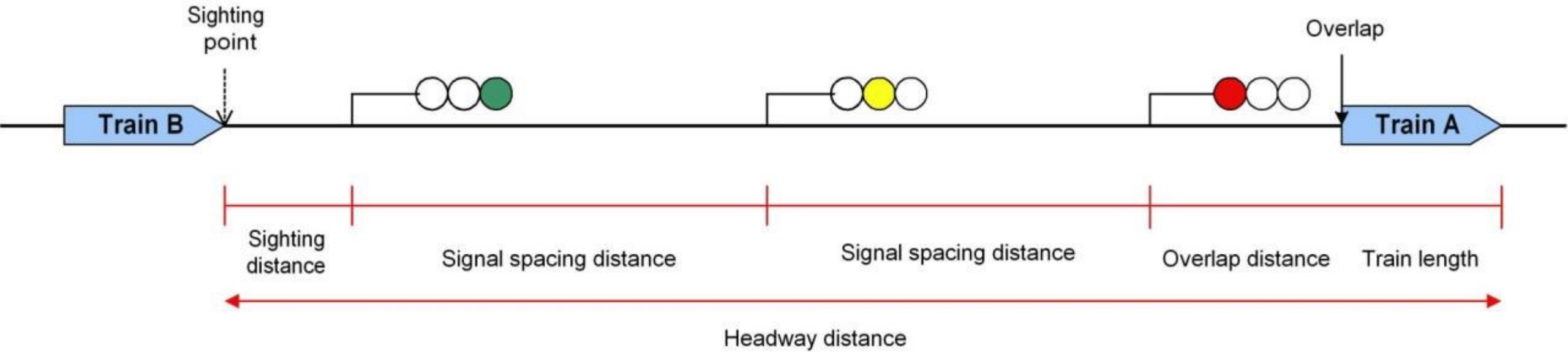
Motor car and train stopping distances compared

Total emergency stopping distances from **80 miles/hour** assuming dry conditions

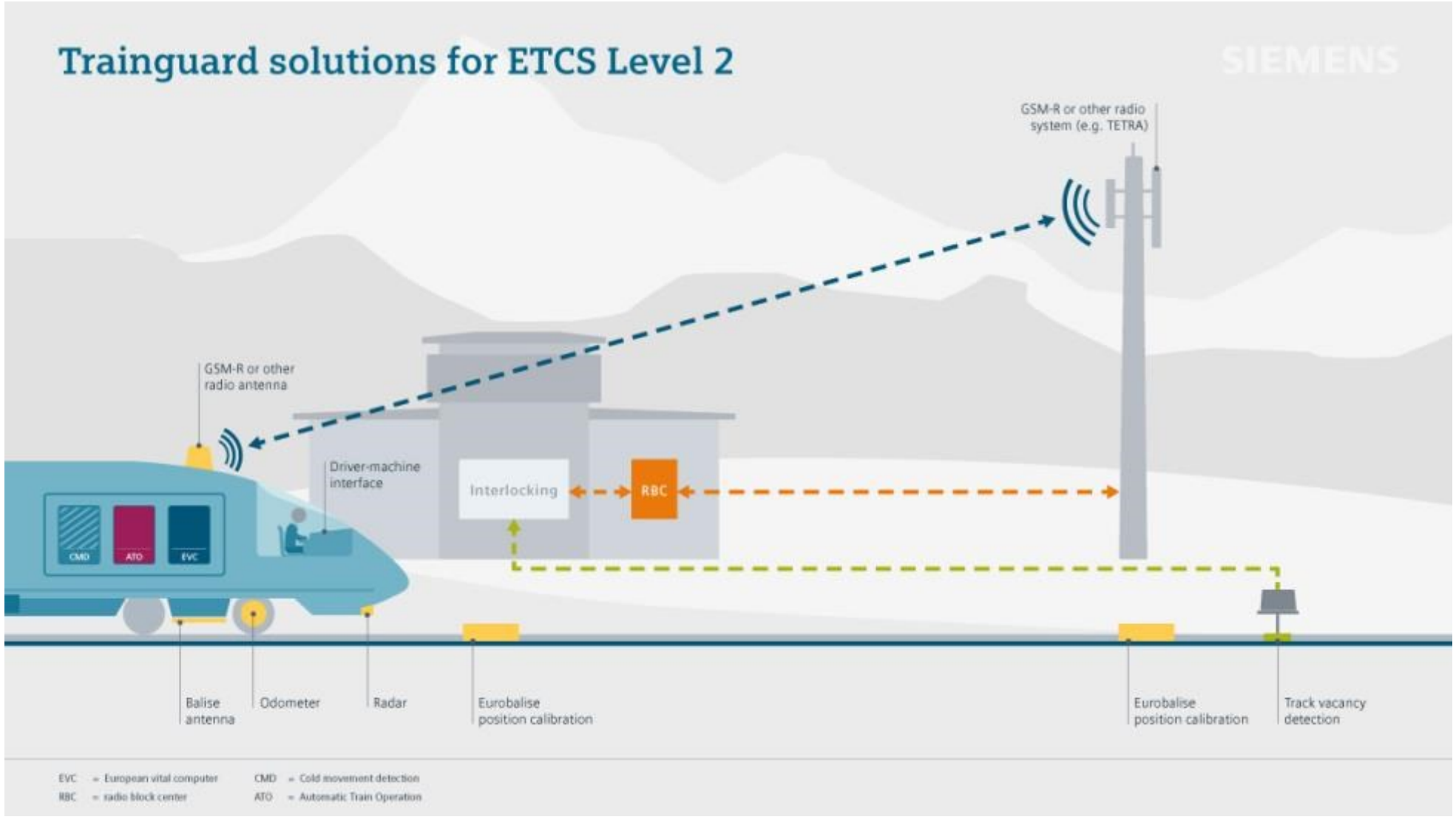


@ 80 miles/hour,
braking distance for train
= 16.7 x braking distance for car

What do we do?

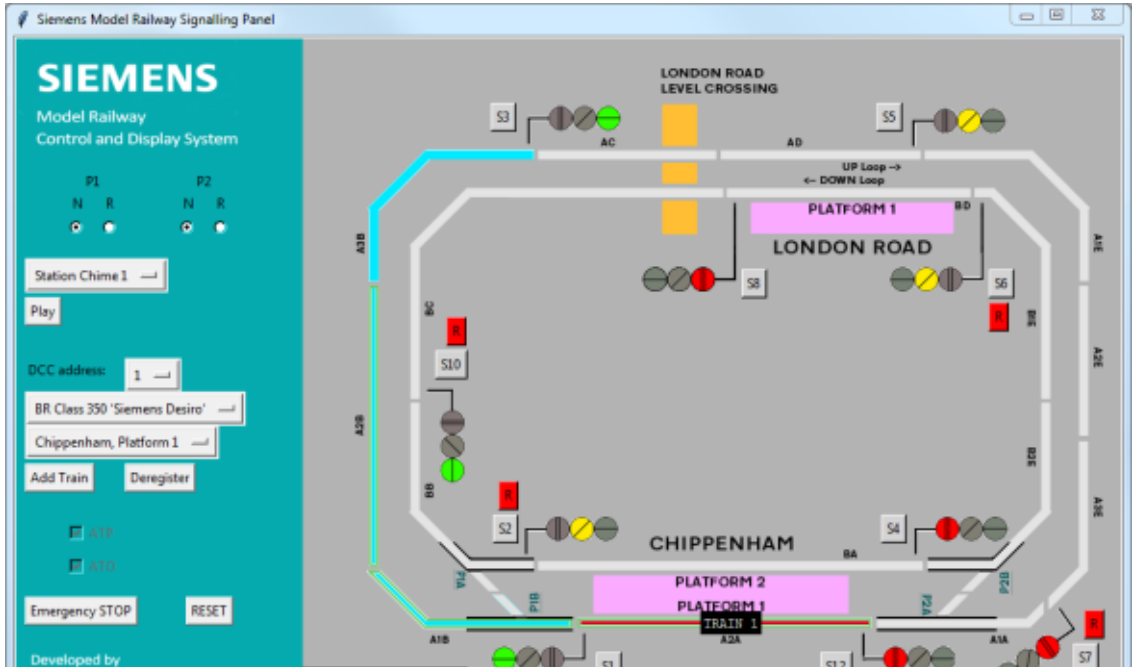

Sighting point
Overlap
Train B
Train A
Sighting distance
Signal spacing distance
Signal spacing distance
Overlap distance
Train length
Headway distance

What do we do?


Trainguard solutions for ETCS Level 2
SIEMENS
GSM-R or other radio system (e.g. TETRA)
GSM-R or other radio antenna
Driver-machine interface
Interlocking
RBC
CMD
ATO
EVC
Balise antenna
Odometer
Radar
Eurobalise position calibration
Eurobalise position calibration
Track vacancy detection
EVC = European vital computer
RBC = radio block center
CMD = Cold movement detection
ATO = Automatic Train Operation

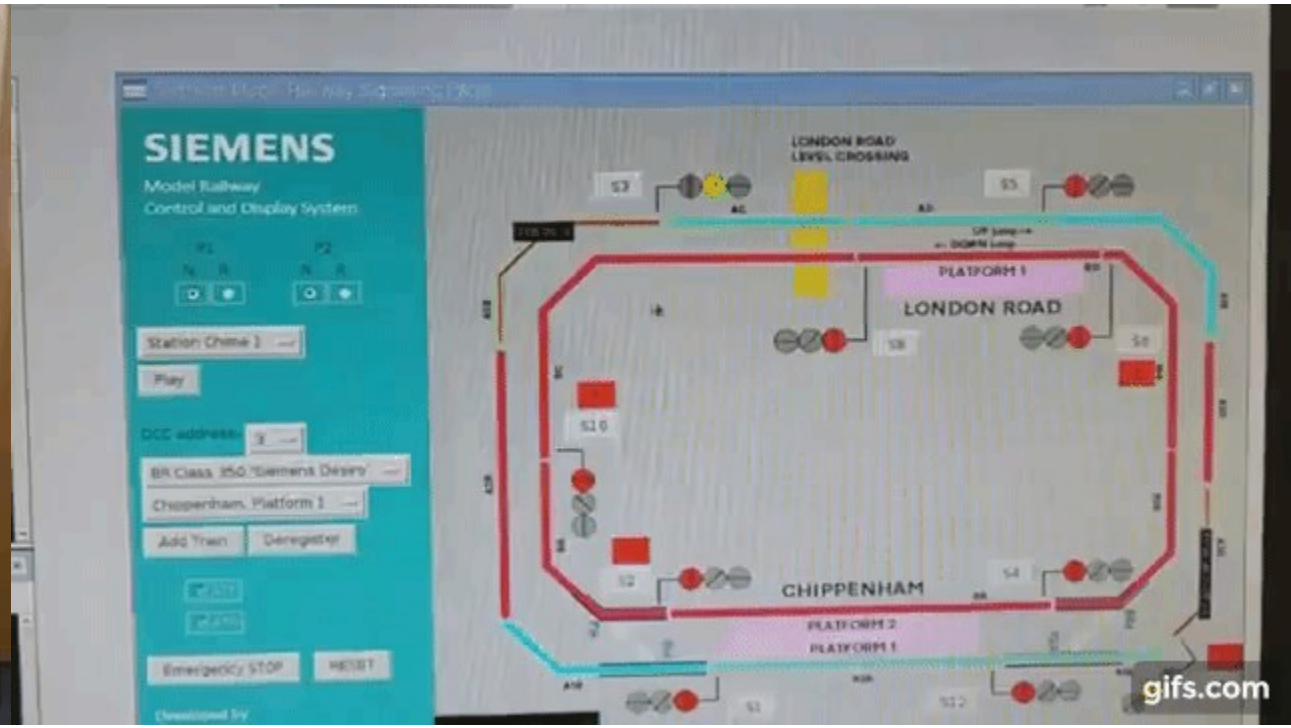
Background

- In 2016, Siemens graduates developed a model railway control system using the Raspberry Pi: **STEMRail**



Background

- **Automatic Train Operation** and **IP Signalling Network**



SIEMENS

What is the STEMRail Model Railway Layout?

- A tool to demonstrate programming, computing, electronics and engineering concepts to young people using technology that is readily available to schools and students.

- Realistic enough to demonstrate railway signalling principles to all ages, yet simple enough to engage young children.

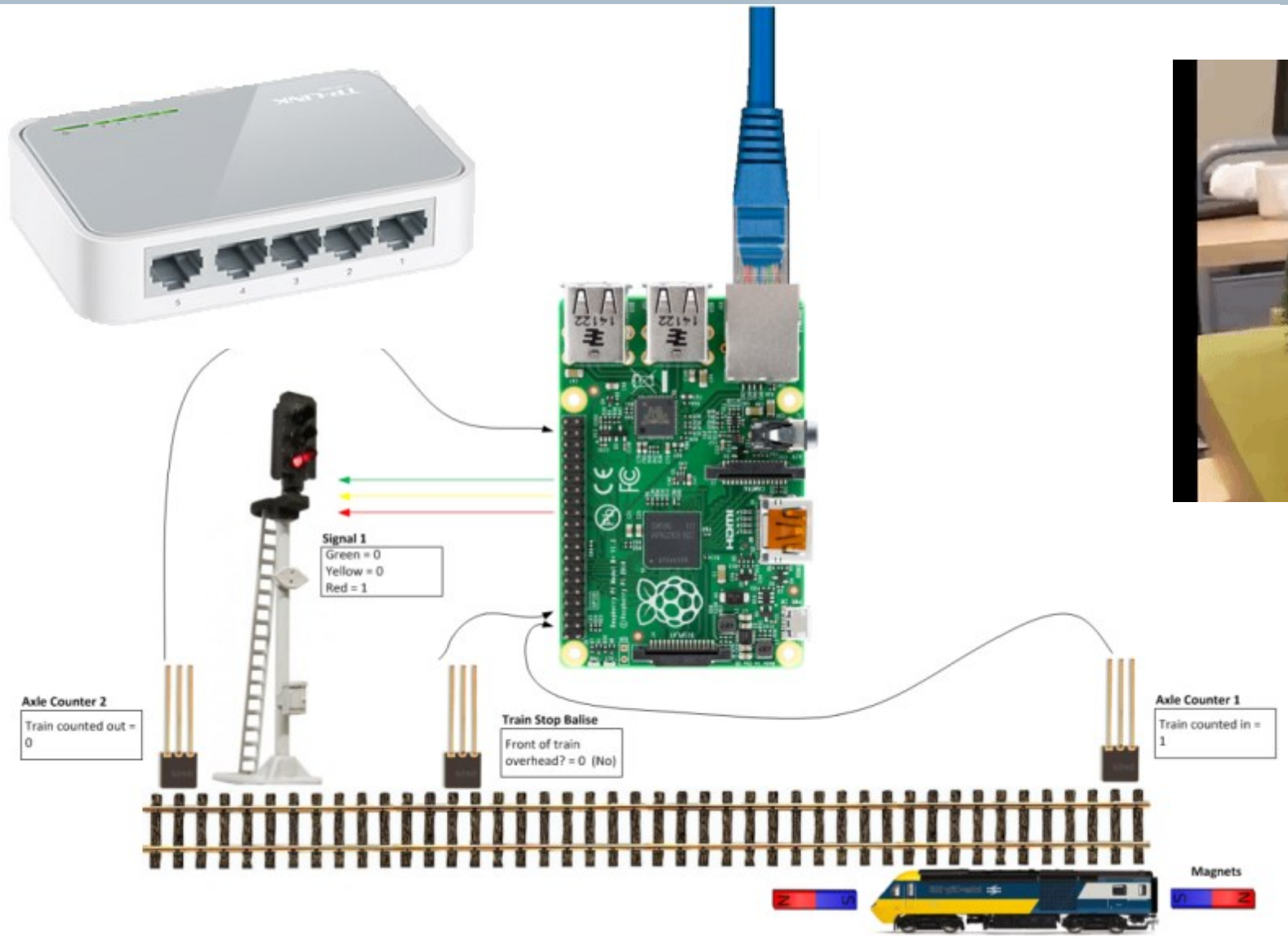
  - **Programming** – Uses Python language as taught in schools

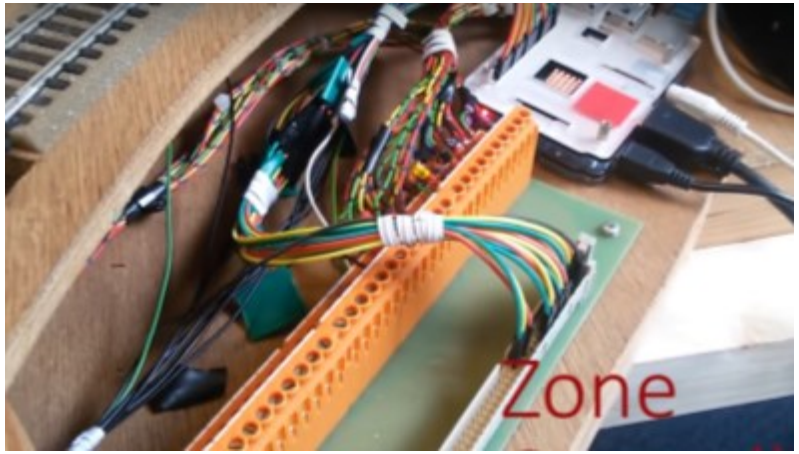
  - **Computing** – The Pi are networked using a simple text protocol

  - **Electronics** – Simple Hall effect sensors and LEDs represent the trackside signalling equipment

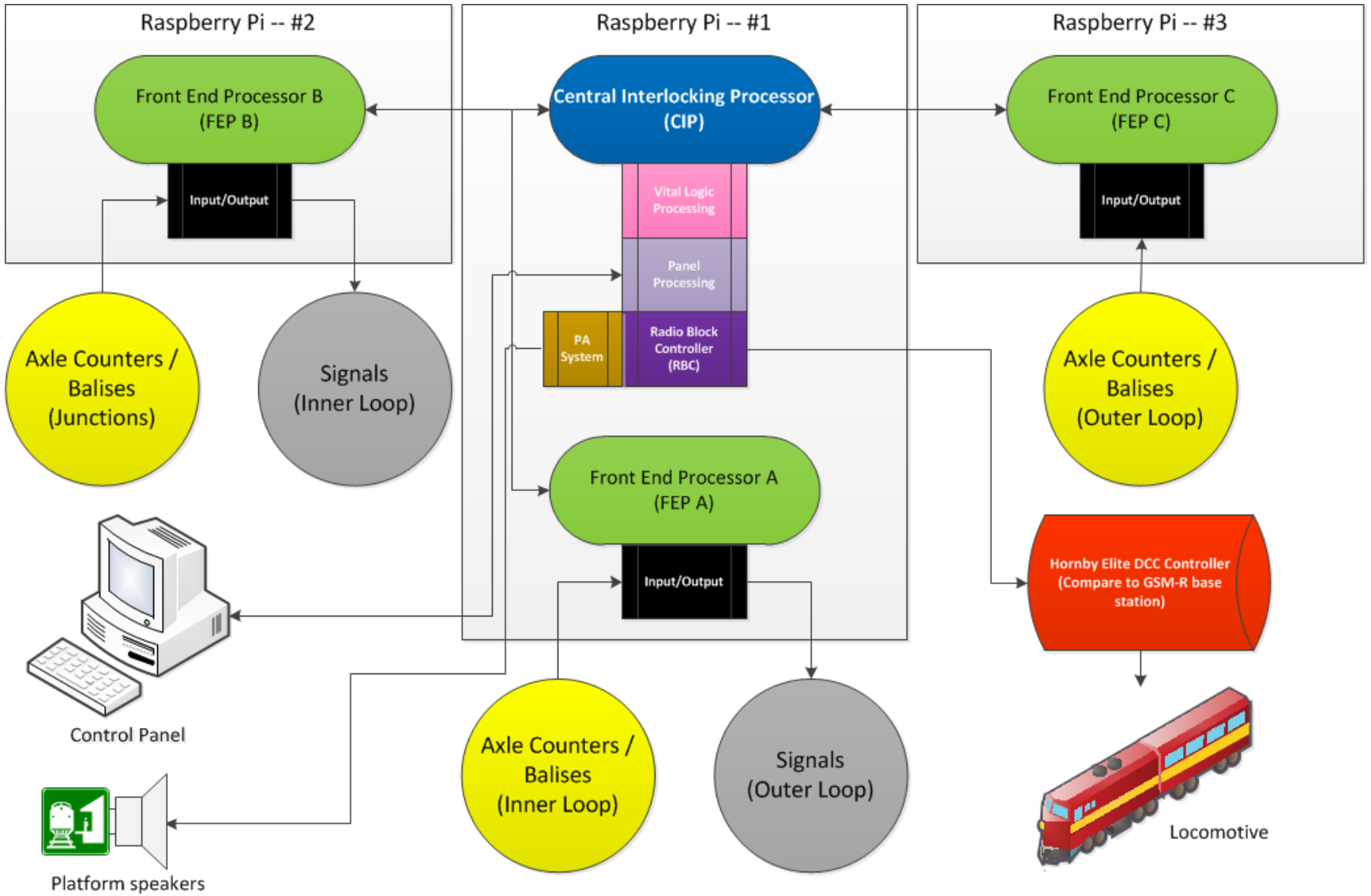
  - **Engineering** – Problem solving, system design and physics such as train braking

STEMRail Hardware





System Architecture



SIEMENS

STEMRail Software

SIEMENS

Control Systems and Object Oriented Programming

Computer programs are made up of a **data** and **algorithms** that manipulate that data.

Data

Algorithms

The greater the **throttle** and lower the **gear**, the greater the **force** pushing the car forward.

$$f = m \cdot a$$

Force is divided by the car's **mass** to determine **acceleration**.

Acceleration multiplied by **Time elapsed** = **Speed**

LAWS OF PHYSICS

STEMRail Object Oriented Control – Data

Track Section Objects

Defined Attributes:
- Track ID Number (location)
- Speed Limit
- GIF images

Variables:
- Occupied / Clear
- Routed / Not Routed
- ID# of Train in Section

Points Objects

Defined Attributes:
- Point ID Number
- Tracks involved
- GIF images
- Button

Variables:
- Normal Lie / Reverse Lie

Train Objects

Defined Attributes:
- Train ID Number
- Train Type

Variables:
- ID# of Current Track
- Direction of Travel
- Target Speed
- Halt Flag
- Emergency Stop Flag

Signal Objects

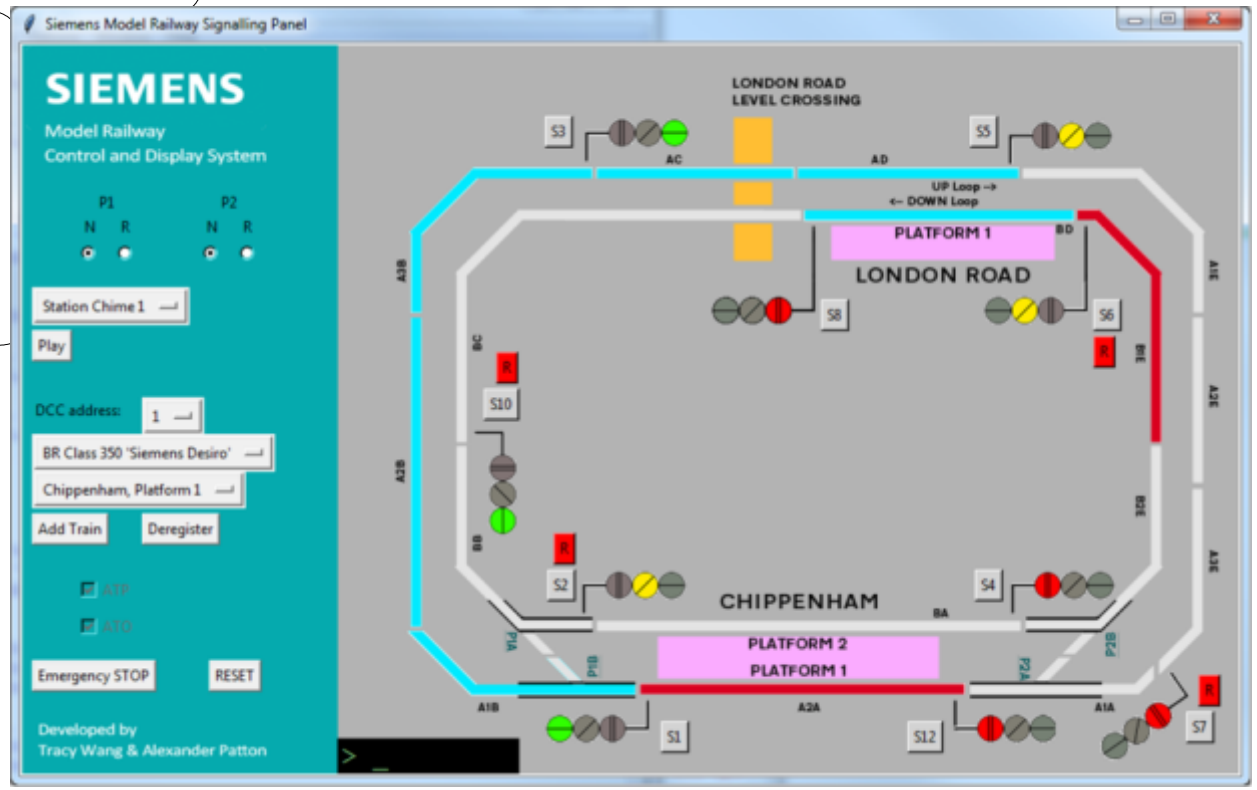
Defined Attributes:
- Signal ID Number
- List of track sections and points being protected
- GIF images

Variables:
- Aspect (Red / Yellow / Green)
- Force Signal to Red (Yes / No)

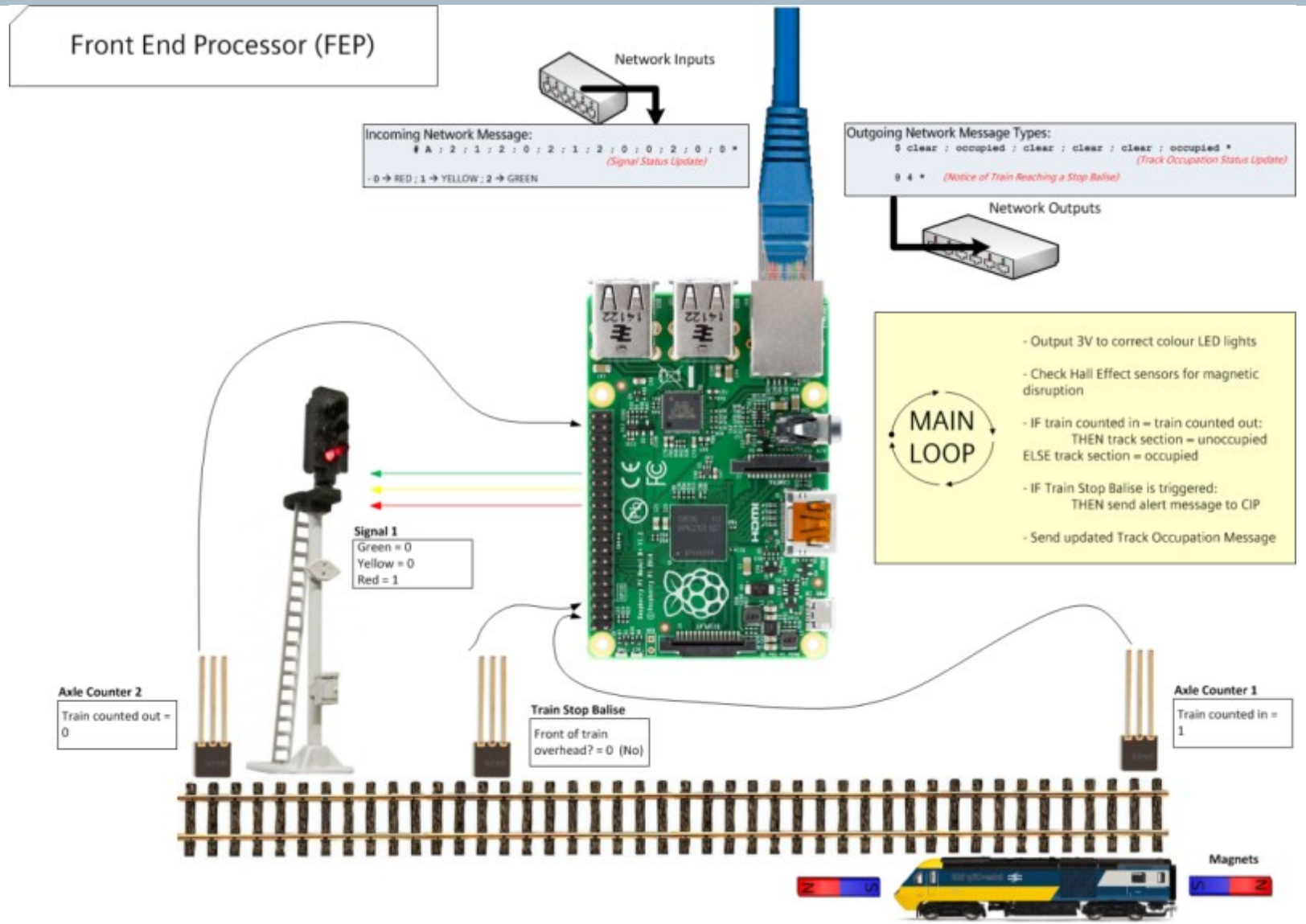
Route Objects

Defined Attributes:
- Direction
- List of Track Sections Covered
- List of Points Required

Variables:
- Set / Not Set



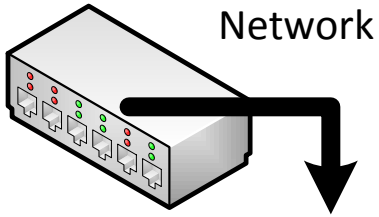
STEMRail Object Oriented Control – Algorithms



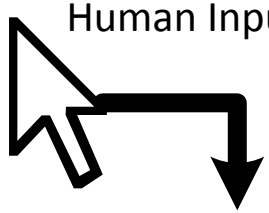
STEMRail Object Oriented Control – Algorithms

Algorithms

Network Inputs



Human Inputs



Incoming Network Message:

`# clear ; occupied ; clear ; clear *`

(Track Status Update)

- Update the state of each Track Object
- FOR EACH Signal Object:
  - IF train in section THEN set signal red
  - ELSEIF next signal is red THEN set signal yellow
  - ELSE set signal green

User clicks:

`Set route from Signal 1 to Signal 3`

- Check if the route is valid
  - next_signal(Signal 1) MUST = Signal 3
  - Signal1.direction MUST = Signal3.direction
- For each Track Object in Route Object, set to Routed if unoccupied

Generate Movement Authority and Calculate Target Speed

For each train: Load the speed limit of the current track section.

Movement Authority Level = 1

Is the next track section clear and routed? Is there no red signal within braking distance?

YES TO BOTH: **Movement Authority Level = 2**

Is the next next track section also clear and routed? Is there no red signal in the next two tracks?

YES TO BOTH: **Movement Authority Level = 3**

Was the trainstop balise in the current track section just triggered? Does MA Level = 1?

YES TO BOTH: **Set the halt flag until MA Level > 1.**

Target Speed = % of Speed Limit

MA Level 1 : 27.5%

MA Level 2 : 75%

MA Level 3 : 100%

STEMRail Object Oriented Control – Algorithms

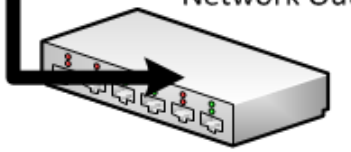
Send Outgoing Network Message:

`# A ; 2 ; 1 ; 2 ; 0 ; 2 ; 1 ; 2 ; 0 ; 0 ; 2 ; 0 ; 0 *`

(Signal Status Update)

- 0 → RED ; 1 → YELLOW ; 2 → GREEN

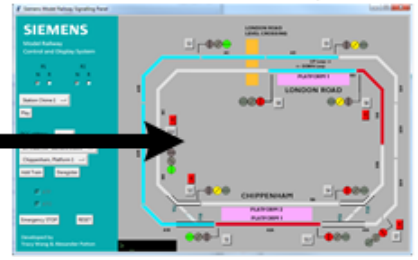
Network Outputs



Display Images on Screen:

```
Signal_1_red.gif ;
Track_A1B_blue.gif ; etc.
```

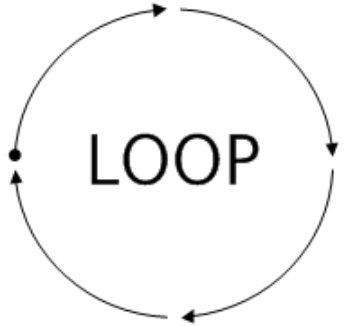
Human Outputs



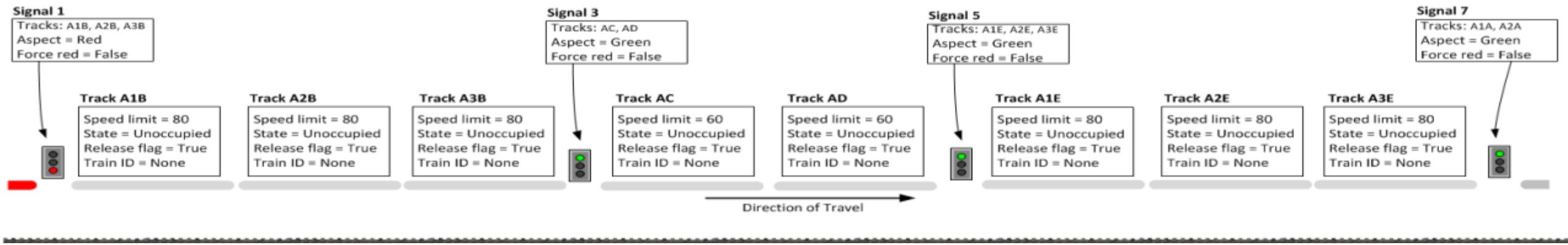
Broadcast Train Speeds:

```
Train 1 set to 60 mph
Train 3 set to 80 mph
```

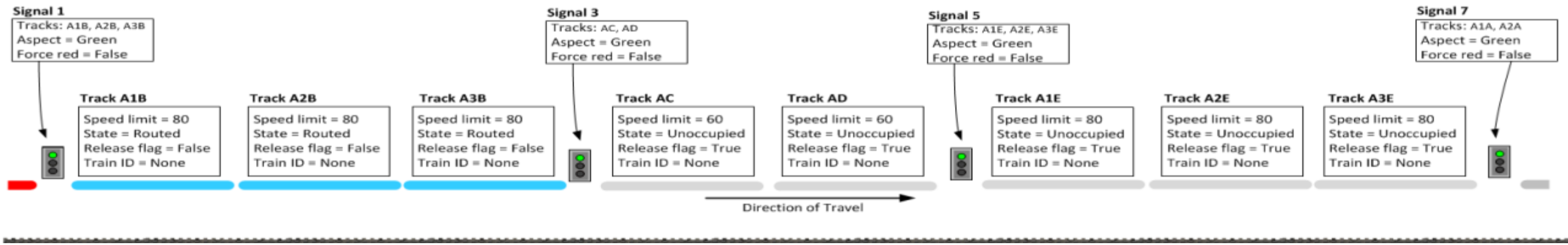
Train Controller Outputs



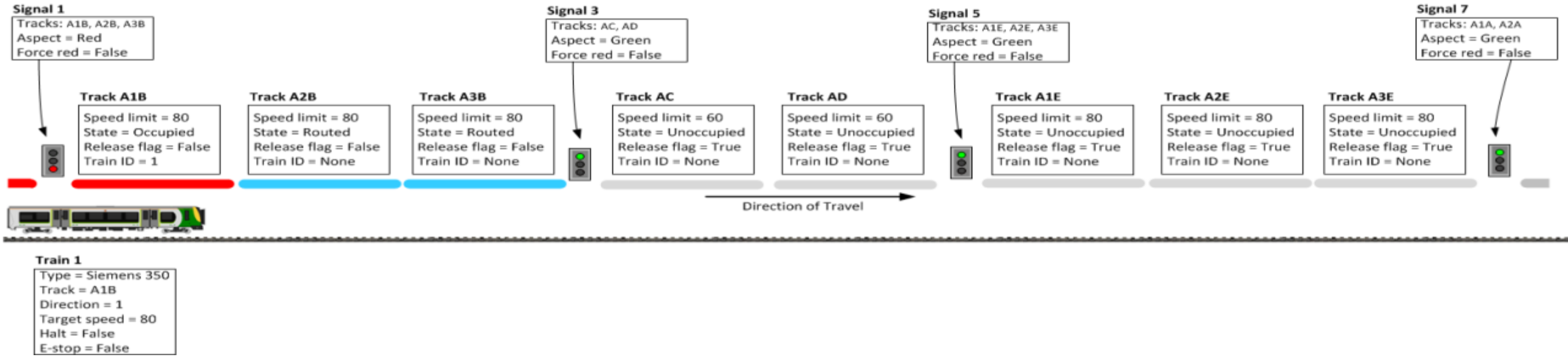
Object Oriented Programming



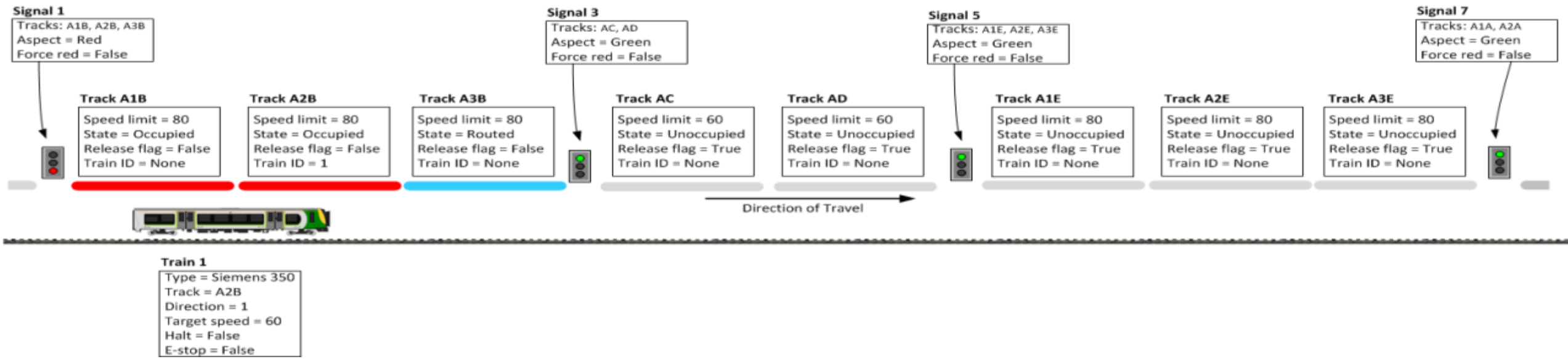
Object Oriented Programming



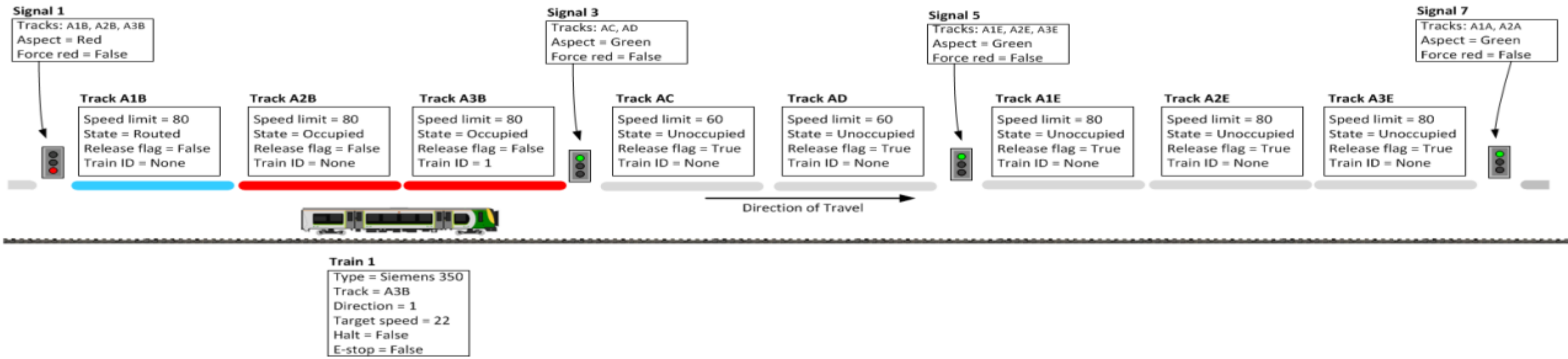
Object Oriented Programming



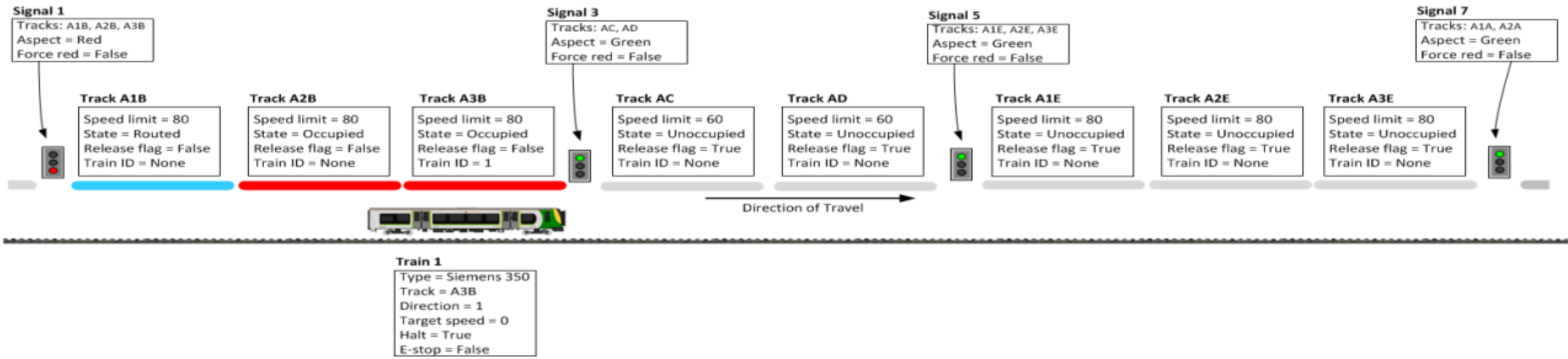
Object Oriented Programming



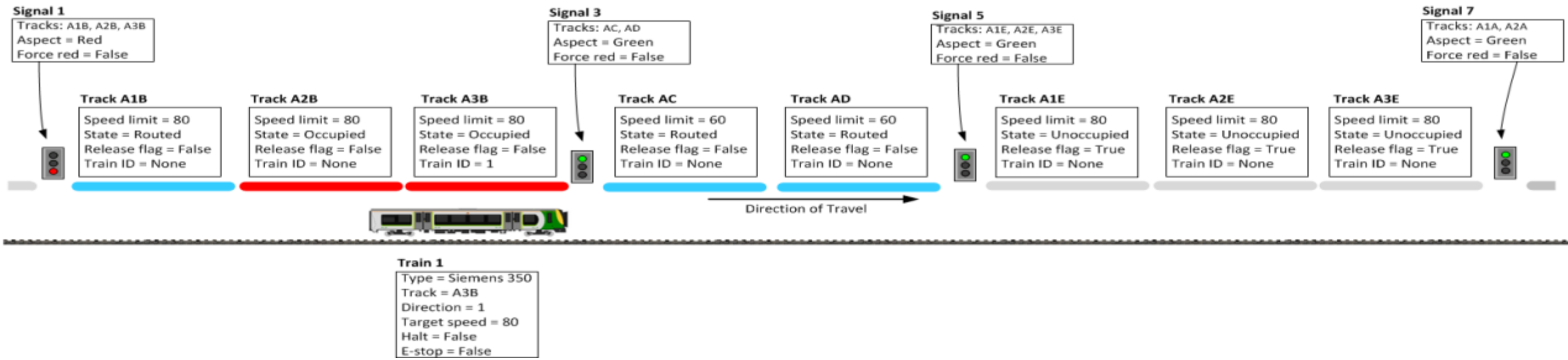
Object Oriented Programming



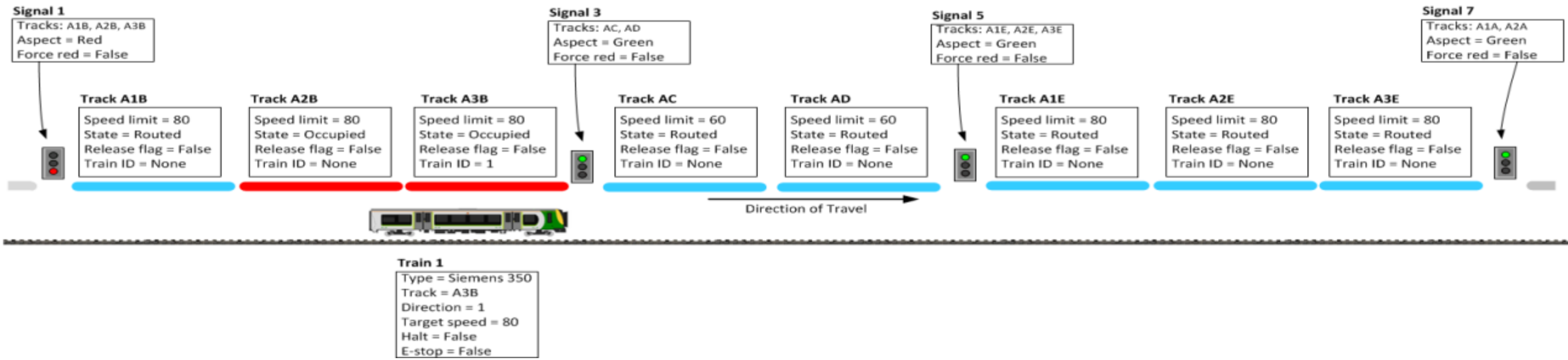
Object Oriented Programming



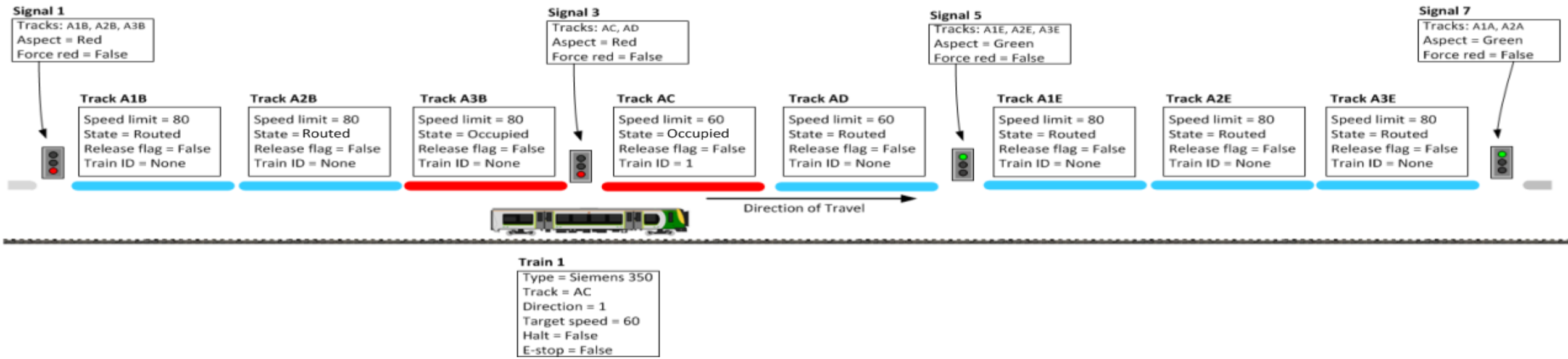
Object Oriented Programming



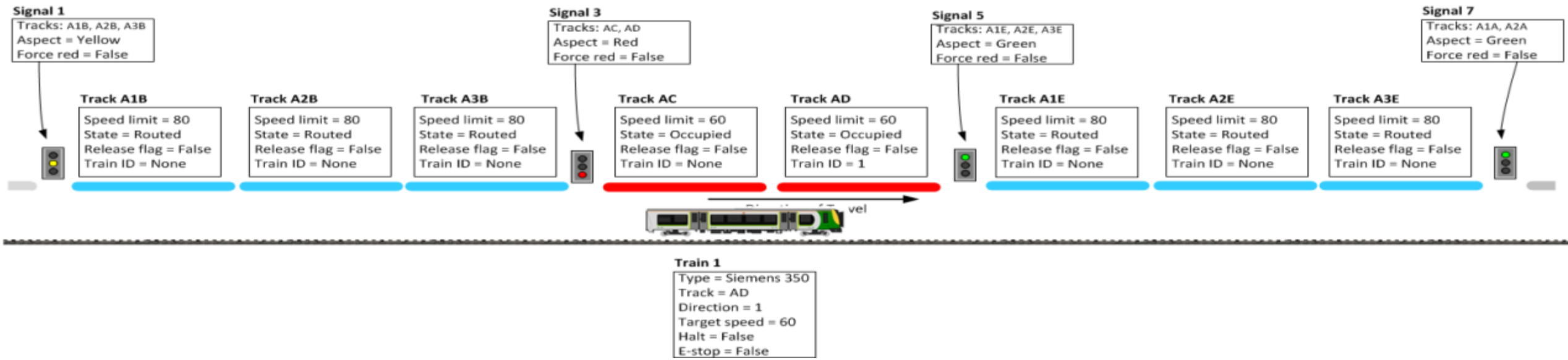
Object Oriented Programming



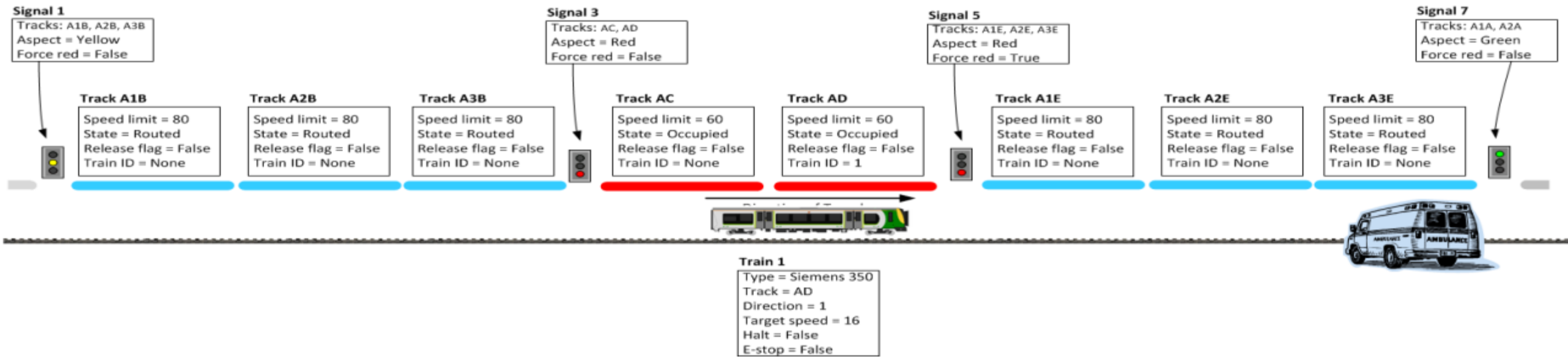
Object Oriented Programming



Object Oriented Programming

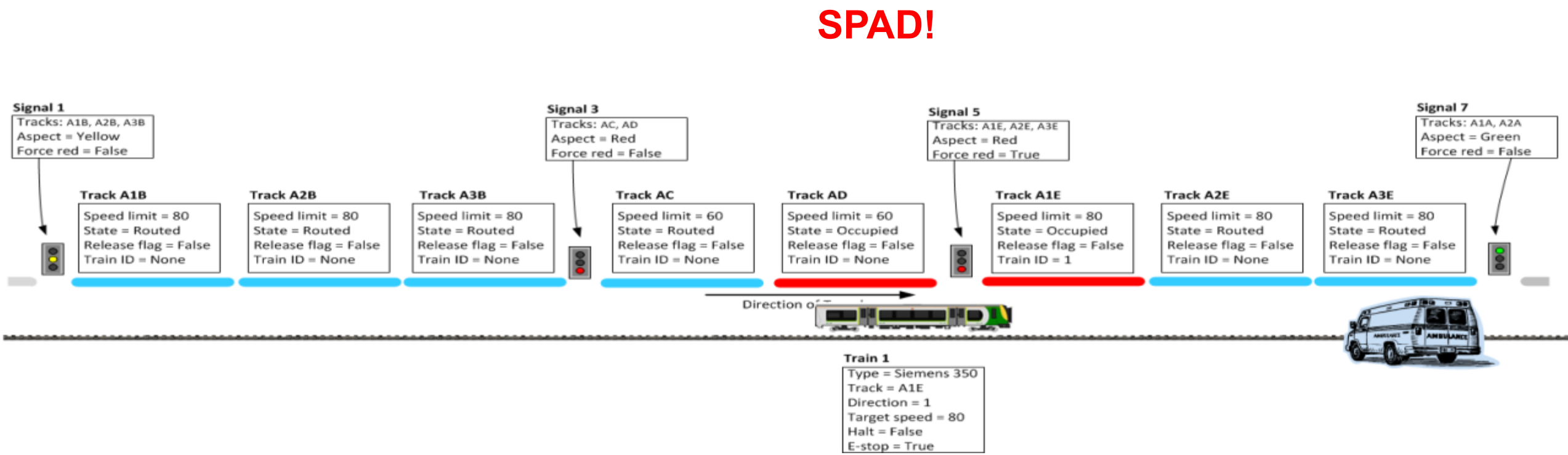


Object Oriented Programming

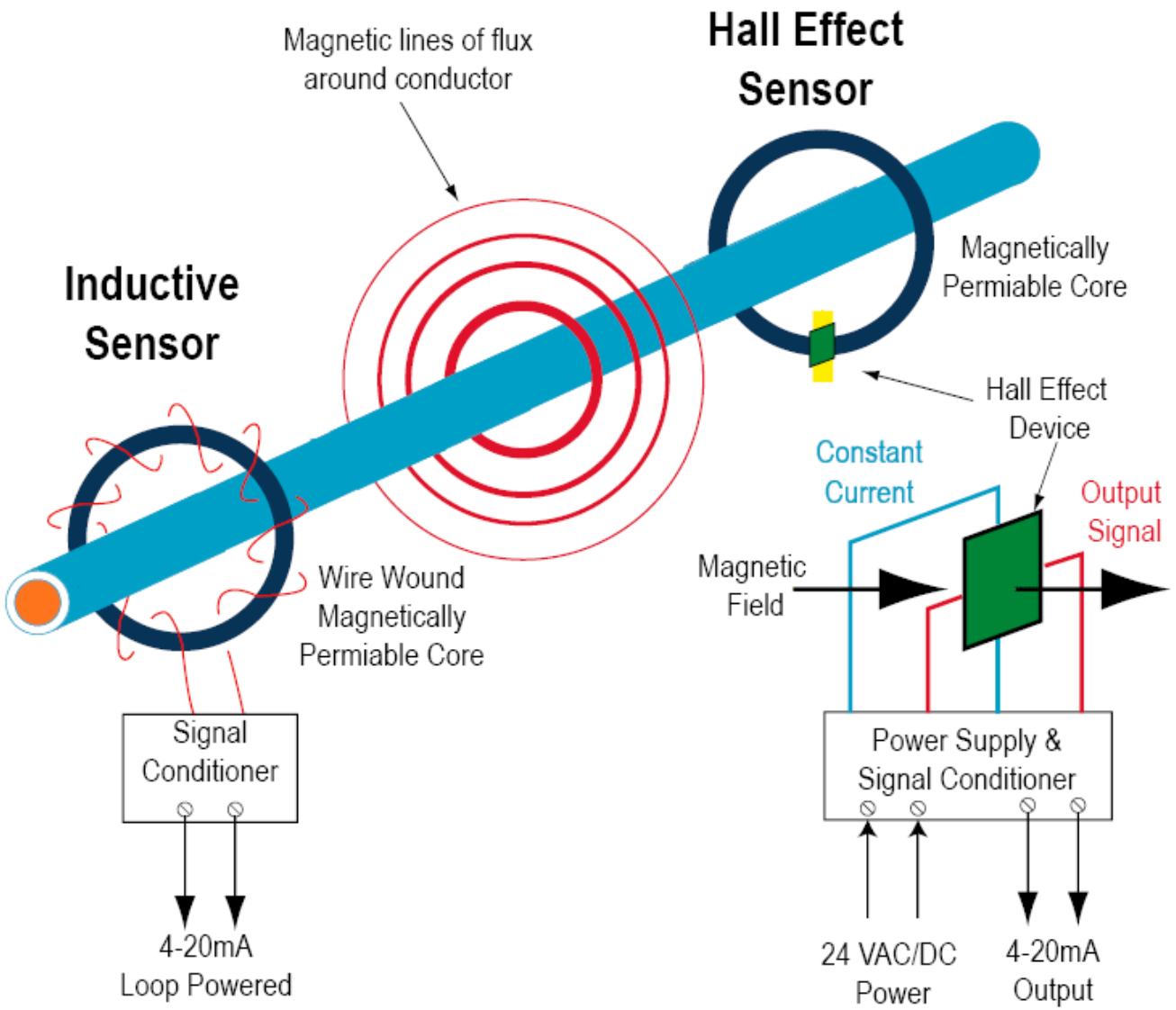


Object Oriented Programming

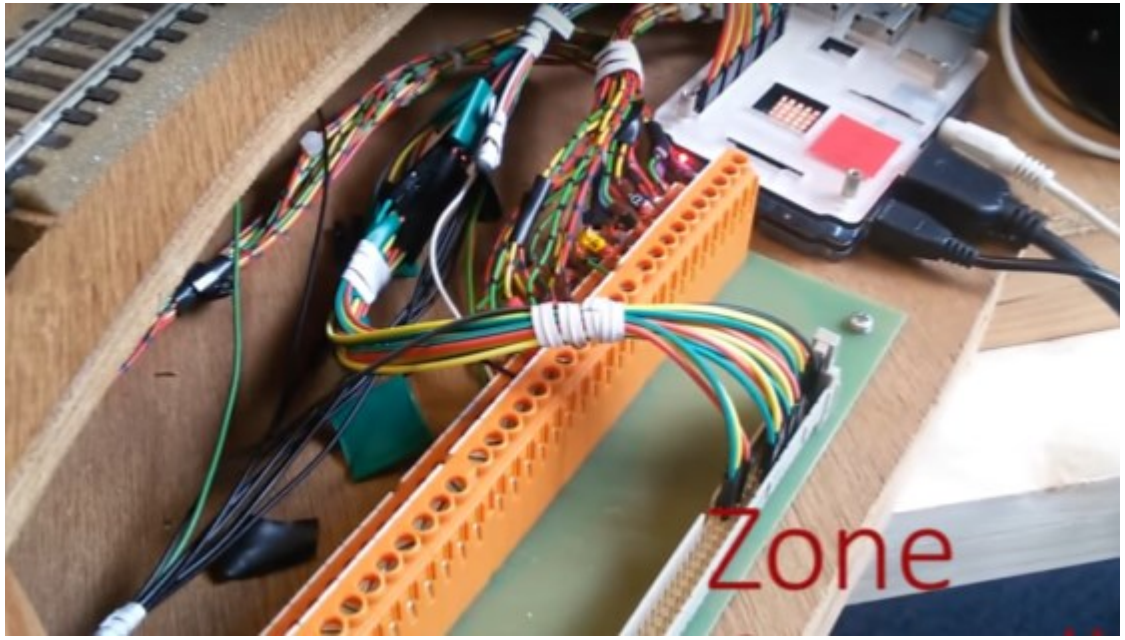
SIEMENS

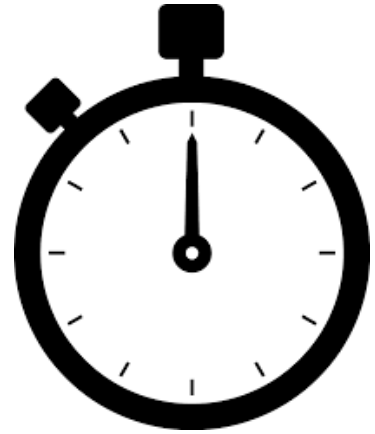


Real World Challenges – Electromagnetic Interference



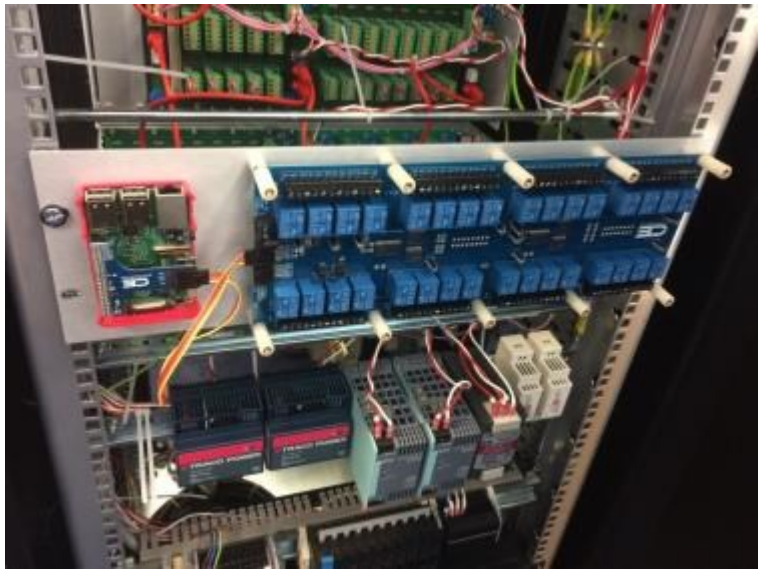
Real World Challenges – Human Factors





SIEMENS

Raspberry Pi – Taking it Further



SIEMENS

Questions

Model Railway Control with a Raspberry Pi

SIEMENS

Thank you

Model Railway Control with a Raspberry Pi