DESN2000 MECH Workshop

Week 7 – Sensors

Class overview

1. **Sensors in everyday life** (5 min)
2. **Analogue and digital** (5 min)
3. **System/circuit diagrams and pseudocode** (15 min)
4. **Report integration** (10 min)
5. **Project time!** (80 min)
6. **Summary** (5 min)

Before we start…

What is a sensor?

What is a sensor?

A sensor in an electronic device, which receives **inputs from the environment** and **outputs data**, which is **used** by an engineering system to **perform a function**.

Sensors in everyday life

Sensors in everyday life	Analogue and digital	System/circuit diagrams and pseudocode	Report integration	Project time!	Summary
5 min	5 min	15 min	10 min	80 min	5 min
All	All	All	All	All	All

Sensors in everyday life

- ❑ You likely have encountered many sensors today!

Activity - 10 minutes

Discuss in your groups, writing down every sensor that you've encountered today as well as what they do. Use the below format.

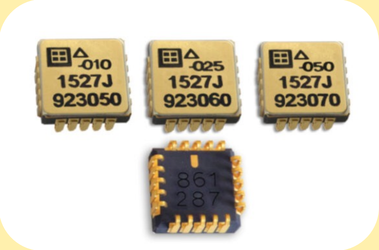
Example: Sydney train capacity sensor

- **Sensor type:** Weight sensors
- **Where:** In the train bogies
- **Data input:** train carriage weight
- **Data output:** A low-medium-heavy indication on TripView about whether my train is full or not

❖ When choosing/brainstorming sensors for your project, try to always take notes considering these four points. They will help you narrow down what sensors are most suitable for your use case, and can form the basis for your report on them.

Sensors in everyday life

❑ Further examples of sensors



Accelerometers

Found in your phone to track your steps

Potentiometers

Records a numerical value with a set range to control room light brightness

Buttons on the tram

Records on/off pressed state, to detect if someone wants to open the door

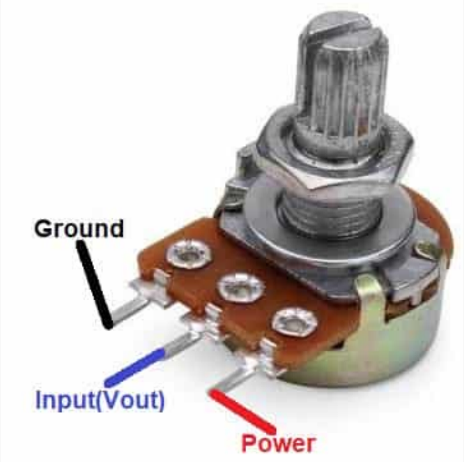
Automatic door sensor

Measures the change in infrared radiation to detect when a person is at the door.

Analogue and Digital

Sensors in everyday life	Analogue and digital	System/circuit diagrams and pseudocode	Report integration	Project time!	Summary
5 min	5 min	15 min	10 min	80 min	5 min
All	All	All	All	All	All

Analogue and Digital



Difference is in the output!

Analogue

Usually outputs a continuous, time-varying current, which changes in magnitude due to resistance.

Example: A potentiometer, depending on the position of the dial, changes its electrical resistance which in turn changes the magnitude of the current which it outputs.

Digital

Outputs a discrete, binary signal that would show as a square wave on an oscilloscope.

Example: A button has only two states, on or off.

Analogue and Digital

Analogue

Pros:

- Higher resolution
- Great for audio/video transmission

Limitations:

- Vulnerable to noise and interference
- Generation loss (taking a picture of a picture reduces the resolution)

Digital

Pros:

- Less vulnerable to interference
- Can be transmitted long range

Limitations:

- More power demanding
- More bandwidth compared to an analogue system transmitting the same information

The choice of sensor will depend on your use case and what you prioritise as most important. Be sure to justify your decisions.

System/circuit diagrams and pseudocode

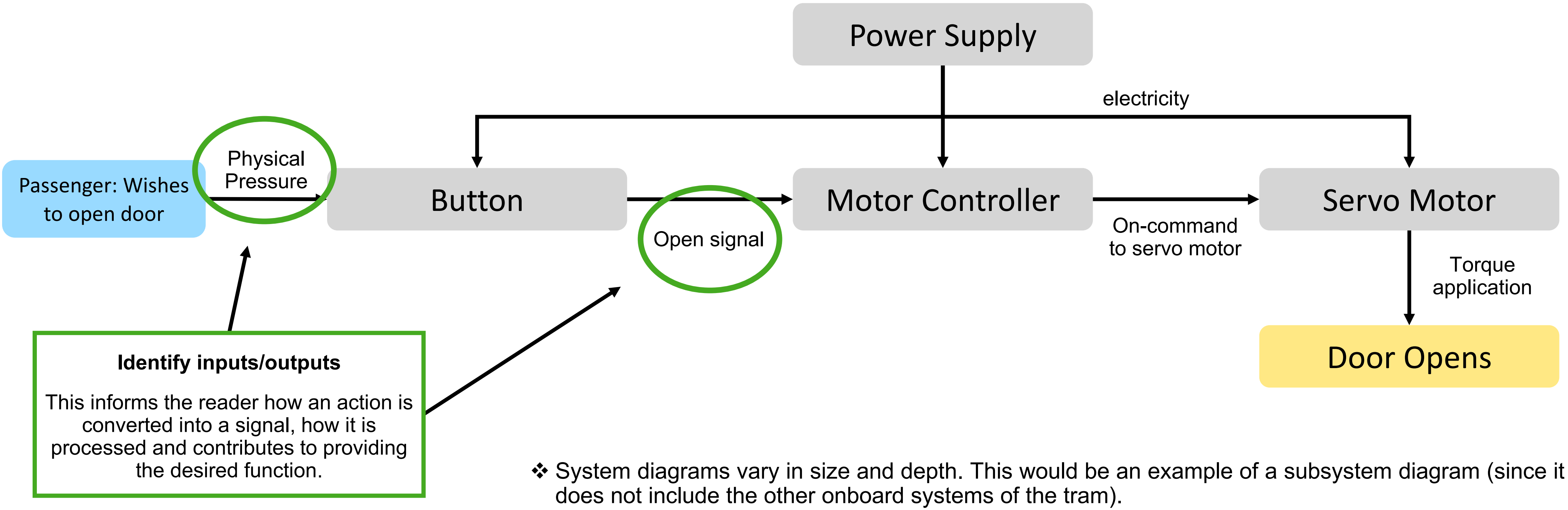
Aka. Things we'd love to see in your report :)

Sensors in everyday life	Analogue and digital	System/circuit diagrams and pseudocode	Report integration	Project time!	Summary
5 min	5 min	15 min	10 min	80 min	5 min
All	All	All	All	All	All

System diagrams

- Systems diagrams provide a high-level view of how information is gathered, processed and transferred between the subcomponents of your device.

Example: Tram door button



- System diagrams vary in size and depth. This would be an example of a subsystem diagram (since it does not include the other onboard systems of the tram).
- If your device has more than one sensor, you may choose to do a systems diagram for each of them, and then combine them all on a single, whole systems architecture diagram to show how they interact as a single unit.

Circuit diagrams

❑ Similar to systems diagrams, circuit diagrams provide an overview of how your components connect to each other, from a wiring perspective.

Example from past report

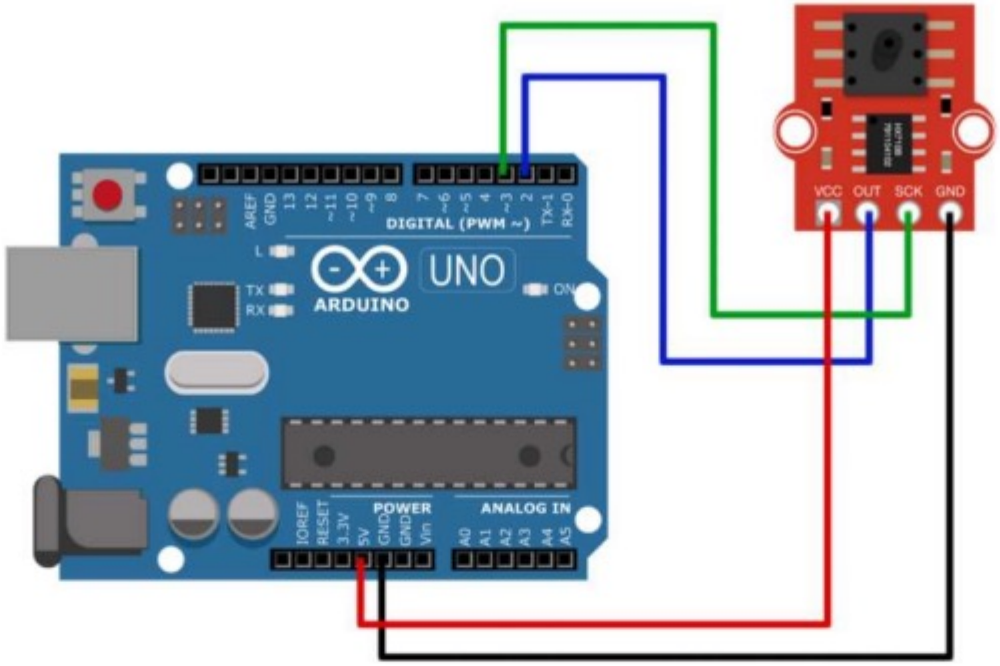


Figure 23. High level circuit diagram of MPS20N0040D Sensor and Arduino Uno

❖ Aim to make one for each of the sensors you analyse

Some software you can use to make these diagrams include:

❑ **Fritzing** (Shown here): **~ $14 AUD to download**. Has both pictorial (illustrated here) as well as traditional black and white line diagram capability. link

❑ **TinkerCAD: Free.** More limited parts selection, though includes the ability to write Arduino code and simulate it. link

❑ **Circuit Diagram: Free**. Online web-based drawer with large variety of components (contains Arduino too). link

❑ **PowerPoint: Free**. If none of the above have the features you need, you could also create these diagrams by downloading top down views of the components you want, and use the line tool to draw the wires. It might not be as quick, but it will get the job done :)

Pseudocode

Purpose: To describe how your sensor/s and processing component/s receive, interpret and output data from a programming perspective. It does so at a reduced level of fidelity such that someone who is limited in programming knowledge can still decipher the method in which your code intends to work.

Example: Accelerometer to detect steps taken by a person

Assume a person takes a step once every second. Therefore, within one second, there should be an “up” and then “down” acceleration as the footstep goes up then down.

Begin by resetting the step counter to 0.

Check if accelerometer is plugged in. If so:

Request person stays still for calibration. Store this value as ”resting” state.

While the power is on and repeating every 1 second:

Read value of accelerometer. Store this value as “before” state.

Wait 0.5 seconds.

Read value of accelerometer. Store this value as “after” state.

If “before” < “resting” and “after” > “resting”:

Increase the daily step counter by 1.

Else:

Continue with the loop.

Just like regular code, use comments to explain your pseudocode as well.

Despite being more “casual” than formal coding language, still provide enough fidelity such that your system is described at a high level of detail.

Report Integration

Sensors in everyday life	Analogue and digital	System/circuit diagrams and pseudocode	Report integration	Project time!	Summary
5 min	5 min	15 min	10 min	80 min	5 min
All	All	All	All	All	All

Report Integration

Here's some things to consider if you choose to explore sensor selection

- **Explain and justify your sensor choice:** Many environmental variables can be measured both using digital and analogue sensors. Thus, explain which version you chose and why it suits your needs.
- **Identify any assumptions you make:** If you are struggling to find information to justify your decisions, use the information you have to make a prediction on what is best. Identify that your decision is based on an assumption in your writing.
- **Make sure to include circuit drawings, system drawings and pseudocode for all sensors you cover in your analysis.**
- **Try to make the diagrams tidy!** They don't need to be picture perfect, but minimise the number of crossing over wires and lay your components out in a logical fashion.
- **Be specific!** If you can, always try to get the actual manufacturer and model names of the components (e.g. if you use an Arduino, specify what model).
- **Be logical!** Phrase your writing to tell a cohesive and concise story of what sensors your device comprises of and what they do, using system/circuit diagrams and pseudocode to aid in your explanation along the way.

Project time!

Sensors in everyday life	Analogue and digital	System/circuit diagrams and pseudocode	Report integration	Project time!	Summary
5 min	5 min	15 min	10 min	80 min	5 min
All	All	All	All	All	All

Summary

Sensors in everyday life	Analogue and digital	System/circuit diagrams and pseudocode	Report integration	Project time!	Summary
5 min	5 min	15 min	10 min	80 min	5 min
All	All	All	All	All	All

Summary

Today we covered:

- ❑ Sensor selection

Next week:

- ❑ Power transmission