

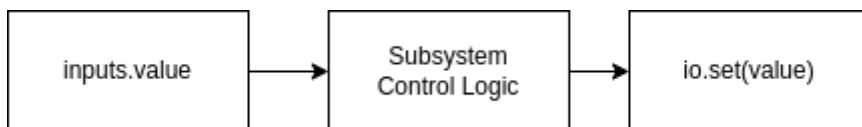
Your First AdvantageKit Subsystem

What is a Subsystem

In FRC one of the end goals of the software is often for the robot to interact or move something in the physical world. To do this we want to split our robot into smaller pieces to make it easier to write and predict what the software will do. Therefore a **subsystem** is a pair of *hardware* and *software* that is responsible for a small part of the robots function. For example the claw from last year was a subsystem that was responsible for its single motor that allowed it to intake and score game pieces.

Components of a Subsystem

A subsystem has hardware that we would like to control and get data from. We are only ever doing 2 things with a subsystem: getting **inputs** or setting **io outputs**.



Inputs

The inputs object is an object that every subsystem has and is responsible for storing the inputs from the system. This could be sensor values, motor states, ect. The inputs object is updated once every loop cycle by the line in the periodic() function.

```
io.updateInputs(inputs);
```

Anywhere else in our code we get values from the inputs object by accessing the data directly using the syntax `inputs.DataWeWant`

For our intake claw we would like to read the value of the line break sensors to tell if we have a game piece. To get this data we need to access it from the inputs object.

```
public boolean getIsLinebreakOne()
{
    return inputs.isLinebreakOne;
}
```

IO outputs

once we have decided what we want to do from the inputs we need to actually tell the hardware to do something. This is accomplished through the IO class of the subsystem. The IO class has a series of methods that allow for our user code to affect hardware. They are accessed through the syntax `io.setThingWeWantToDo(Value)`

For example the claw intake from 2023 could set the velocity of its motor through the following code

```
public void setVelocity(double velocity)
{
    io.setVelocity(velocity);
}
```

The IO class handles all of the hardware specific code and all that is needed to provide it the value. Often we also will provide values to the IO class in "real world" units such as meters/second or degrees.

Revision #5

Created 12 July 2023 19:08:33 by jheidegg

Updated 12 July 2023 20:39:49 by jheidegg