

# DLD Group Project

---

Theo Stangebye - March 28, 2018

Group Members: Nate, Daniel, Theo, and Armand

## Project Info

---

Our goal is to design a system that accomplished 3 items for each "classroom" or compatible study area on campus (called classrooms from here out). Each classroom has 1 bit of output and 1 bit of input. Each of these signals is **Active High**. In each classroom, there is an `classroomController` that communicates with systems outside the classroom and interacts with the electrical systems in that classroom. The input and output bits of each `classroomController` controls/monitors its room with these values: `LightsAreOn` , `ProjectorIsOn` , `ClassroomInUse` , `projectorEnable` and `lightsEnable` . All of the `classroomController` s are connected to a `buildingController` which collects their data. Finally, a `campusController` monitors all of the `buildingControllers` and collects information from them.

- There are 4 buildings on campus that we want to integrate into the system.
- Each building can have up to 64 classrooms, each with its own `classroomController` .
- Each building has a `buildingController` that collects information on the state of each classroom.
- There is a `campusController` that reads information from each of the `buildingController` s.

## `classroomController`

Each `classroomController` has 3 sensors: `LightsAreOn` , `ProjectorIsOn` and `ClassroomInUse` and two things that it controls: `projectorEnable` and `lightsEnable` . The controller also has a TX bit, an RX bit, and 6 `roomID` bits to communicate with the rest of the building. Think of the `roomID` bits as select bits that allow the `buildingController` to select which classroom it is talking to. There is also a `clock` input on each `classroomController` .

The `classroomController` a 6 bit binary number associated with it, these 6 bits are the controllers `roomID` . The `classroomController` should do the following:

1. If someone begins using the classroom - or if `ClassroomInUse` = 1, the controller should make sure that `projectorEnable` and `lightsEnable` are active and remain so until the classroom is no longer in use.
2. Read from the classroom's sensors ( `LightsAreOn` , `ProjectorIsOn` and `ClassroomInUse` ) and store their values into memory.
3. When the classroom's controller is selected by a `buildingController` (meaning that the `buildingController` has broadcasted this specific `classroomController` 's unique 6 bit number on the `roomID` bits), the `classroomController` should use its `RX` and `TX` bits to transmit it's sensor data to the `buildingController` . We will use a FSM to recognize sequential bit strings on the `RX` and `TX` pins to communicate.

## buildingController

The building controller should increment through all the classrooms for a building, staying on each classroom for 10 clock cycles. For instance, starting at the first classroom in the building, it should transmit '000000' to all of the `classroomController` s in the building using its `roomSelect` output (this is 6 bits that are wired to the `roomID` selector on the `classroomControllers` ), wait one clock cycle, and then transmit a "00100010" on its `TX` bit and store the 8 bits from its `RX` pin to an 8 bit bit-string before incrementing `roomSelect` to '000001' and repeating for the second classroom. **The `buildingController` also provides**

the clock signal that is used by all of the `classroomController` s.

For this week let's focus on getting the main functionality of the `buildingController` implemented and we can focus on interfacing with the `campusController` and storing all 64 classroom's worth of information later. For now, just store the bitstream from each classroom into one 8 bit number and overwrite that number for every classroom.

A bit-string is a concurrent number of bits (think a number of wires in parallel) while a **bitstream** is a sequential series of sequential bits sent across 1 wire.

## campusController

The `campusController` switches through the different buildings collecting all of their data just like the `buildingController` switches through all of the classrooms in a building. It should store all of this information into onboard memory.

We enable multiple `controllers` to communicate on the same bus by setting their `TX` and `RX` bits to a HighZ state when it is not selected on its network.

