# Lecture 11 - Digital Comms

Communication is key!

# But first…

# Course Review (System Design)

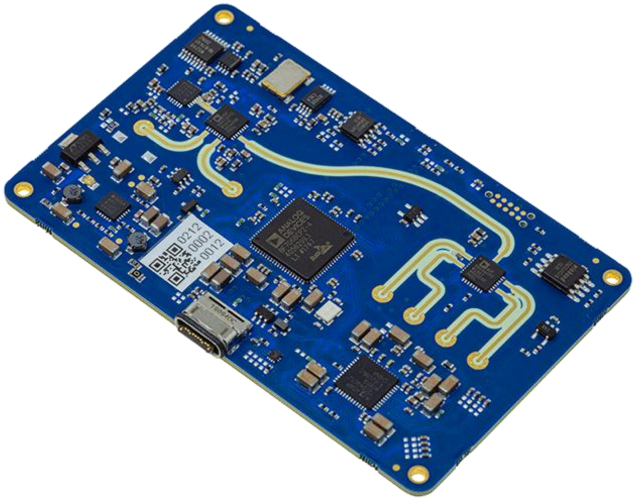
# System Design

THE ART AND SCIENCE OF PCB DESIGN

IAP 2026

**Printed Circuit Boards are medium for fabricating electronic devices**

**PCB Design**



**You just did this!**

@ MIT

# System Design

## Defining System Parameters



**After devising a solution (our electronic system), we need metrics to design/measure it by**

### Electrical Performance

How well the device operates

- Speed/clock rate
- Output power
- Latency
- Resolution
- Accuracy
- Power consumption

Defined by the problem-to-be-solved

### Form Factor

How the user interacts with the device

- Size
- Weight
- Control interfaces
- Training requirement

Defined by the end-user

### Manufacturability

How the device is built

- Cost
- Procurement speed
- Fabrication constraints
- Assembly constraints
- COTS or custom
- Export restrictions

Defined by PCB fab/assembly capabilities

COTS – Commercial Off The Shelf

# System Design

## Modelling the System

[1] M. Dwye and D. S. Ricketts, "The North Carolina State University Rabbit Radar: Build a Frequency-Modulated Continuous-Wave Radar in a Day [Application Notes]," *IEEE Microwave Magazine*, vol. 21, no. 5, Institute of Electrical and Electronics Engineers Inc., pp. 136–145, May 01, 2020. doi: 10.1109/MMM.2020.2971184.

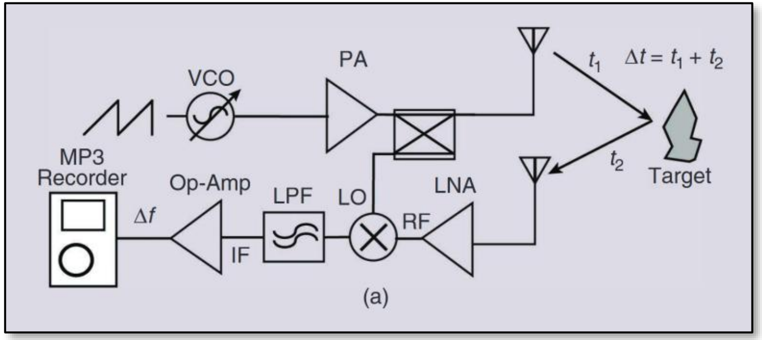
THE ART AND SCIENCE of PCB DESIGN

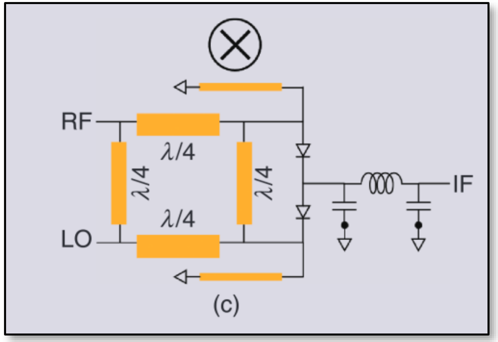
IAP 2026

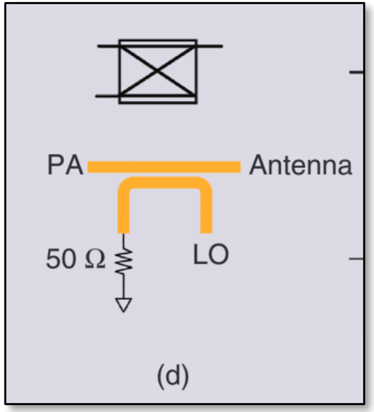
**2-words, 3-syllables:** ***Block Diagrams!***

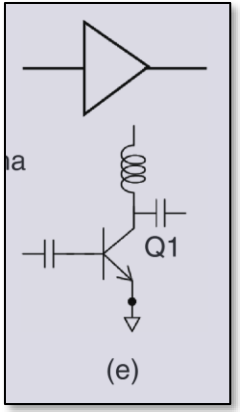
**Abstract the electrical system into lumped components and use circuit interconnections to show the flow of information/power/electrons**

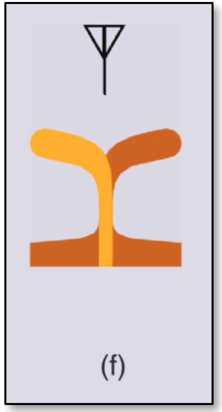
***NC State Rabbit Radar [1]***



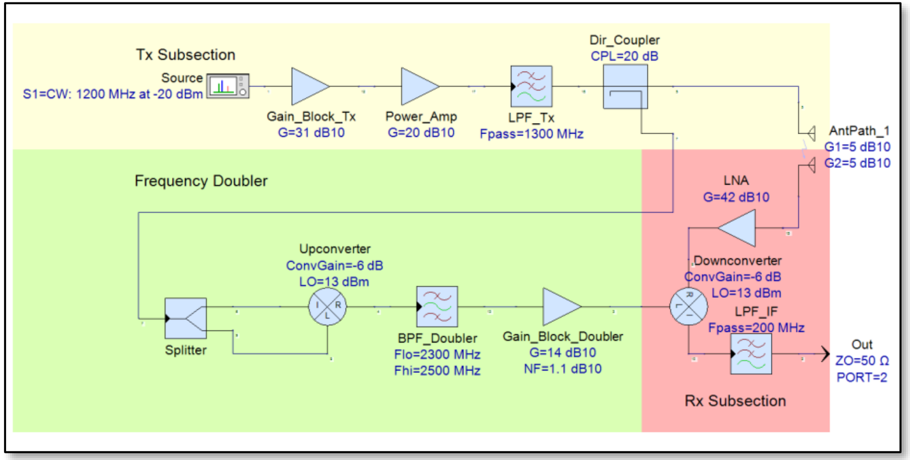

**Breakdown the system into individual components**



@ MIT

# System Design

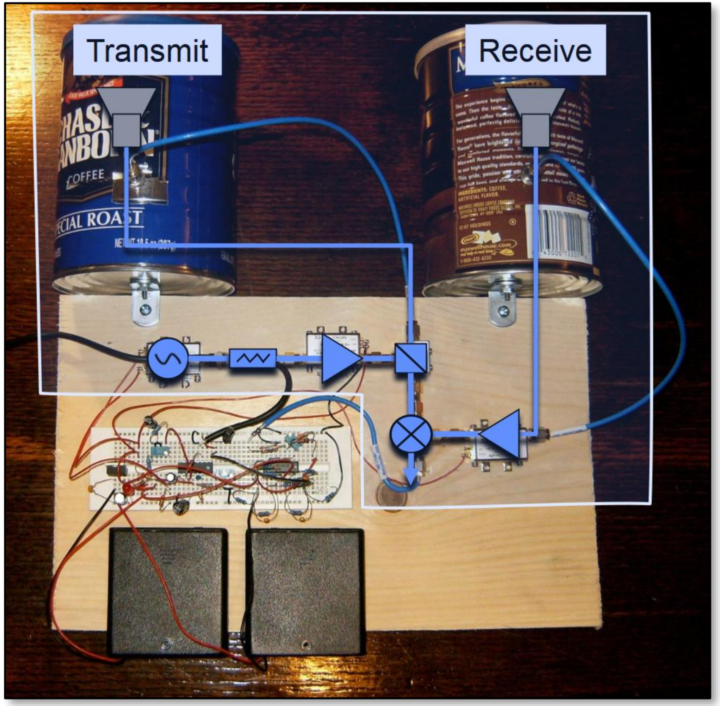
## Modelling the System




**Using our block diagrams we can perform simulations at various levels:**

1. **System level**
2. **Circuit level → We did this using SPICE!**
3. **Component level**

**Block diagrams help explain system functionality at a high level**



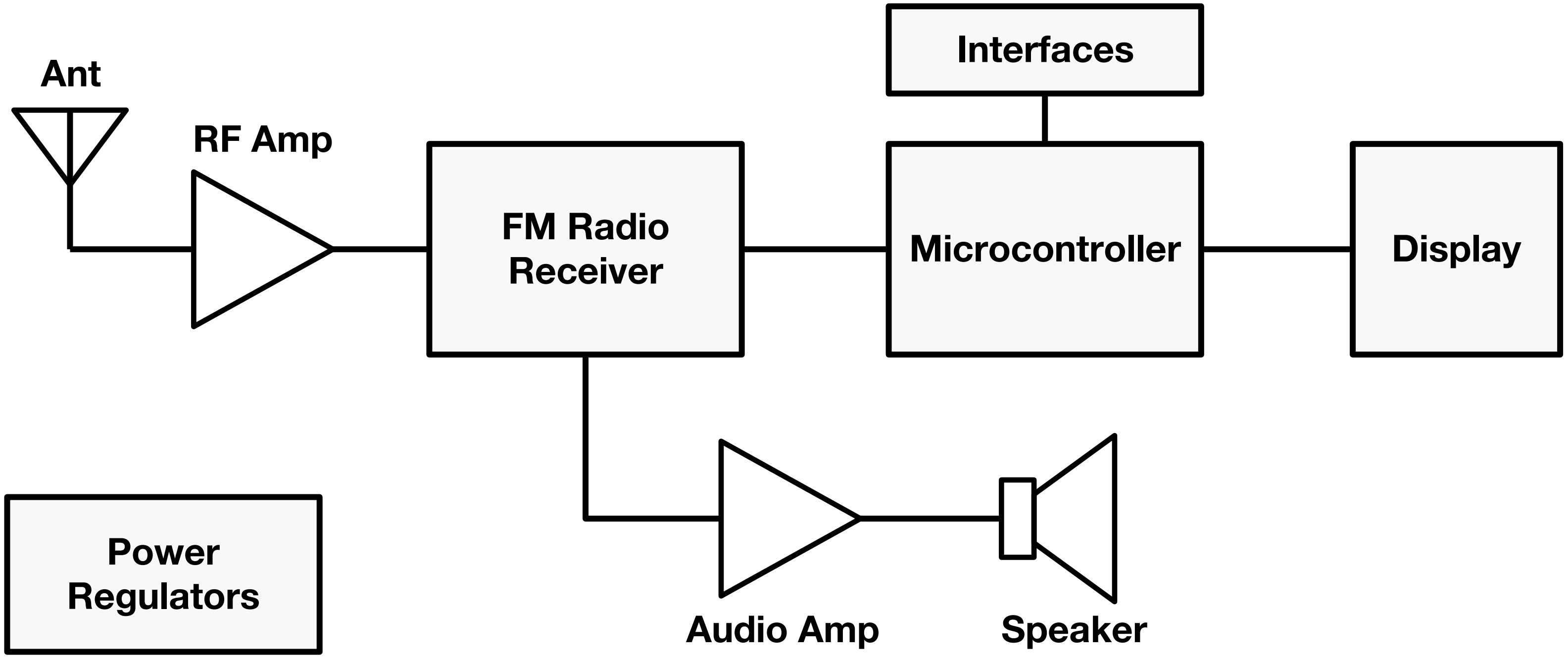

THE ART AND SCIENCE of PCB DESIGN

IAP 2026

@ MIT

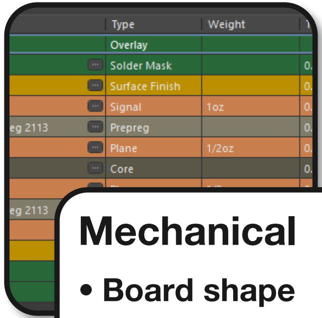
# Schematic

## Block Diagram



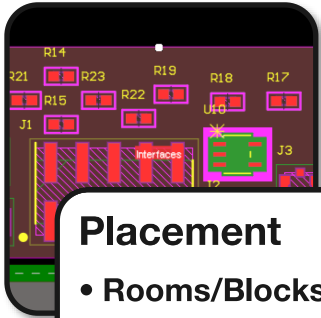
# Basic PCB Layout

## Design Flow



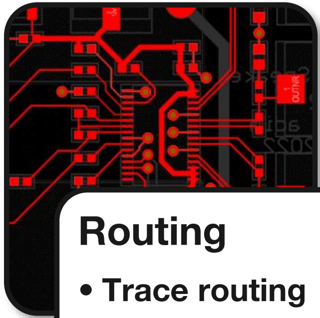

### Mechanical

- Board shape
- Mechanical Features
- Stackup




### Placement

- Rooms/Blocks
- Keep-out zones
- Individual component arrangement



### Routing

- Trace routing
- Copper pours


### Verification

- DRC Report
- Simulation

Repeat as needed

THE ART AND SCIENCE of PCB DESIGN
IAP 2026



@ MIT

# System Design

## Course Project

THE ART AND SCIENCE of PCB DESIGN

IAP 2026

Identify Problem

Devise Solution

Identify System Parameters

Create/Model System

Design PCB! → Fabrication → Test/Debug

FM radio receiver

Course Project!

@ MIT

# System Design

THE ART AND SCIENCE of PCB DESIGN

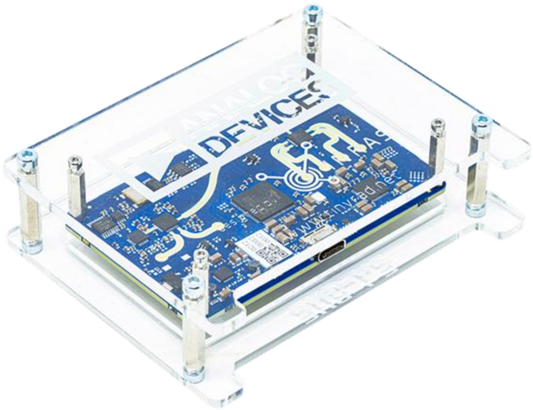
IAP 2026

**Now you're all set to design…**

***Radio Astronomy***

Or your…

- **UROP**
- **Class project**
- **Thesis**
- **Personal pursuit**




***Radar Education Platform***


***Driver Assistance***

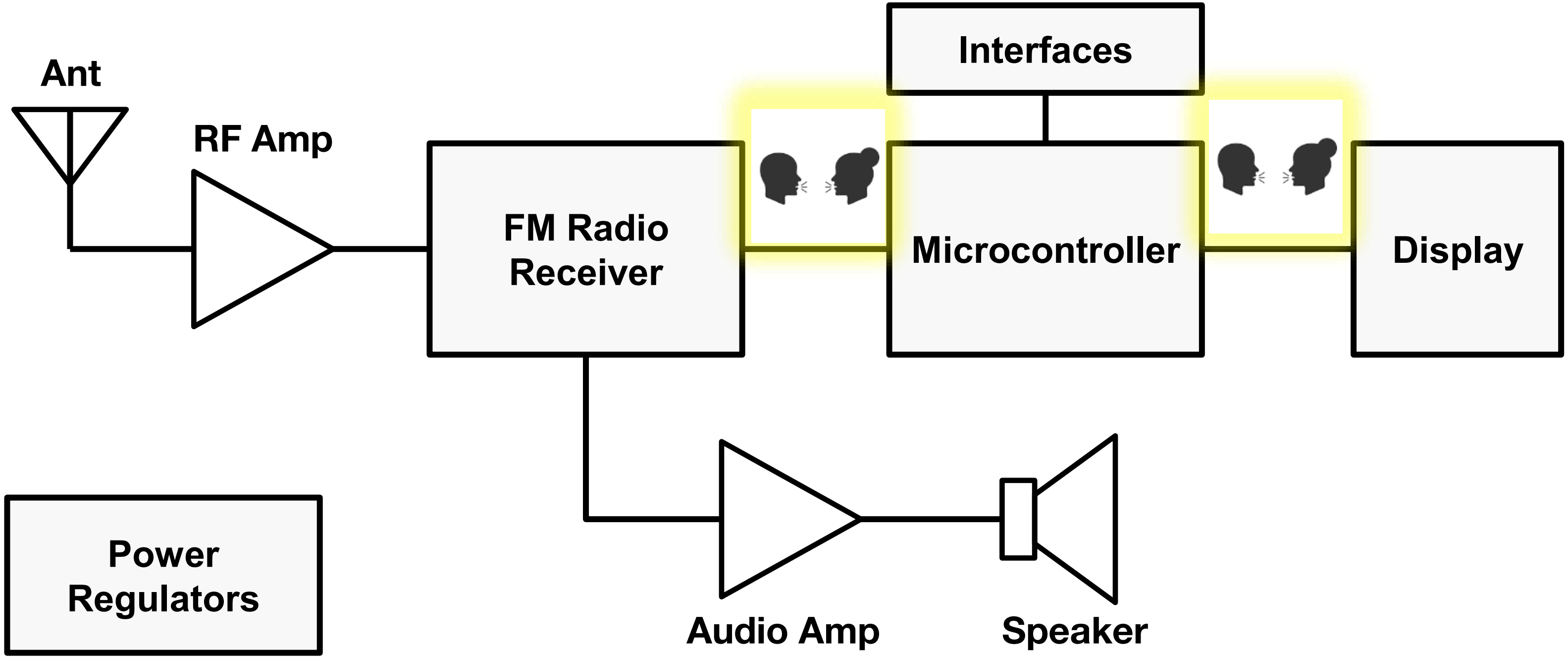

***Terminal Missile Defense***

# Outline

- Introduction
- UART
- SPI
- I2C
- CAN
- Summary

# Schematic

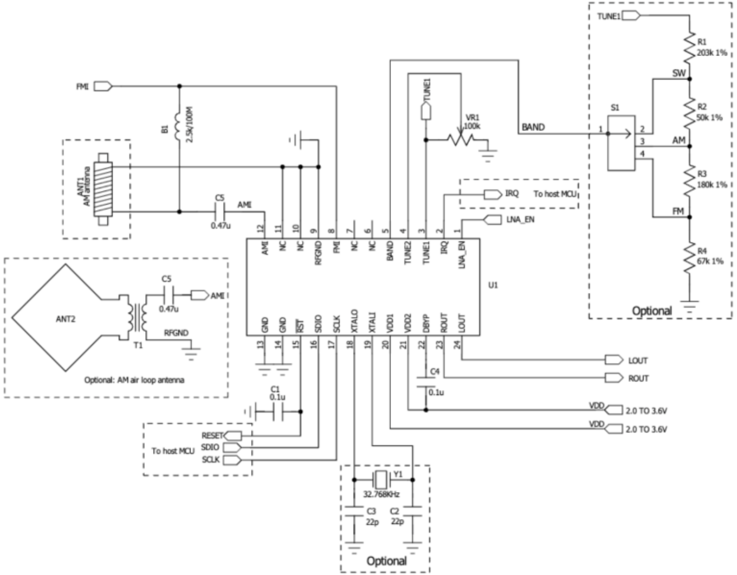
## Block Diagram



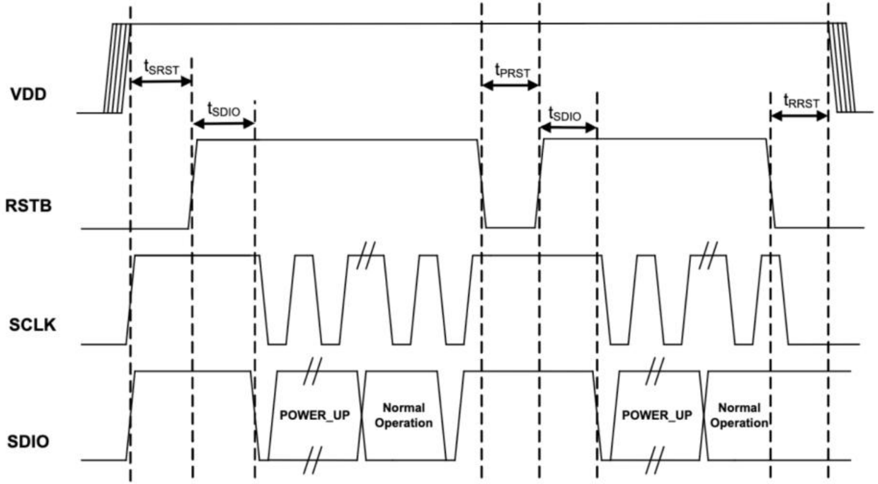
# Datasheet

## FM Receiver Chip

THE ART AND SCIENCE OF PCB DESIGN
IAP 2026



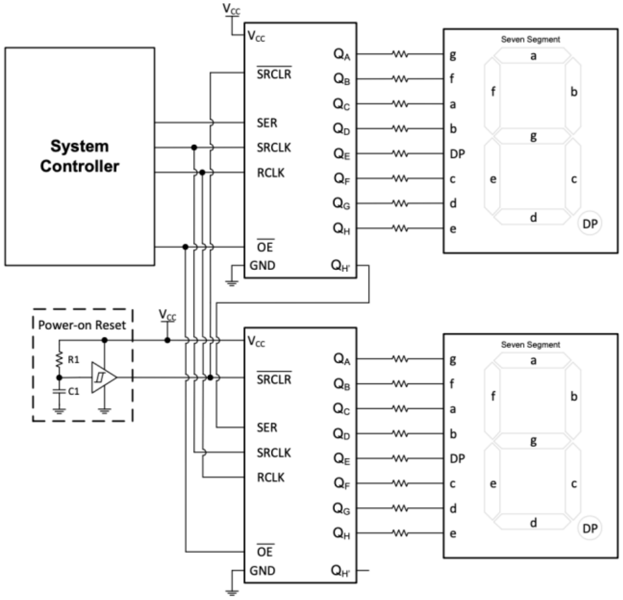

**FM Receiver Chip Typical Application Circuit**



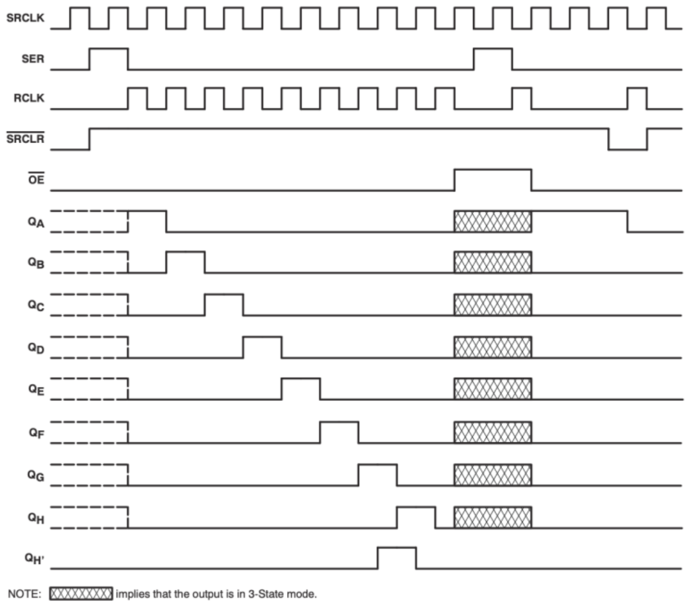

**FM Receiver Chip Reset Timing Diagram**

# Datasheet

## Shift Register




**Shift Register Typical Application Circuit**

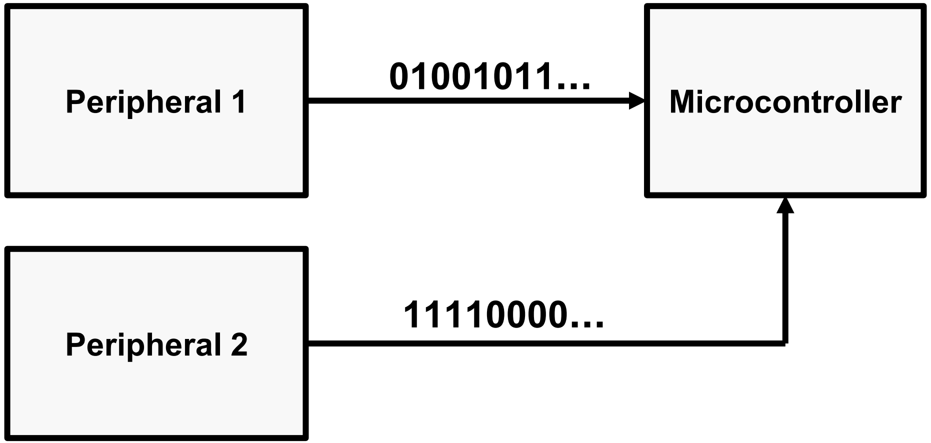



**Shift Register Timing Diagram**

# Communication Protocols

## Why do we standardize communication?

**When our microcontroller and our peripheral communicate, we need to make sure they're on the same page.**




**Let's say we wanted the microcontroller to read the data from peripheral 1.**

**The microcontroller could read:**

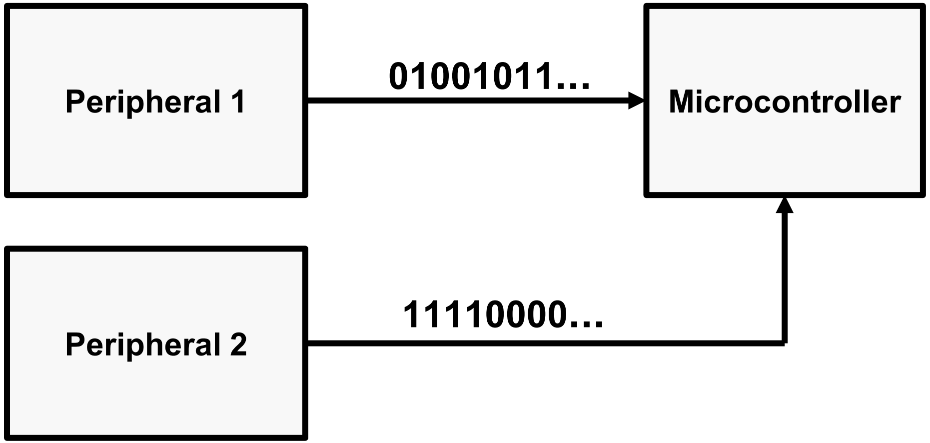
- **01001001 (correct)**
- **00110000 (sampling rate too high)**
- **11110000 (wrong peripheral)**
- **10111111 (switches peripherals)**

**etc.**

# Communication Protocols

## How do we standardize communication?






**What do we want to control?**

- **Transmit and receive rates**
- **Peripheral selection**
- **Data addressing**
- **Device addressing**
- **When to read and write**

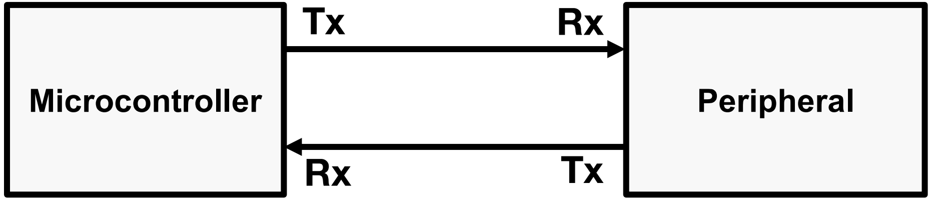
# UART

## Overview

**If we only had two devices, what's the bare minimum we need to ensure that they successfully communicate with each other?**

- **Data path**
- **Shared timing**
- **Start and end indications**

**UART (Universal Asynchronous Receiver/Transmitter)**



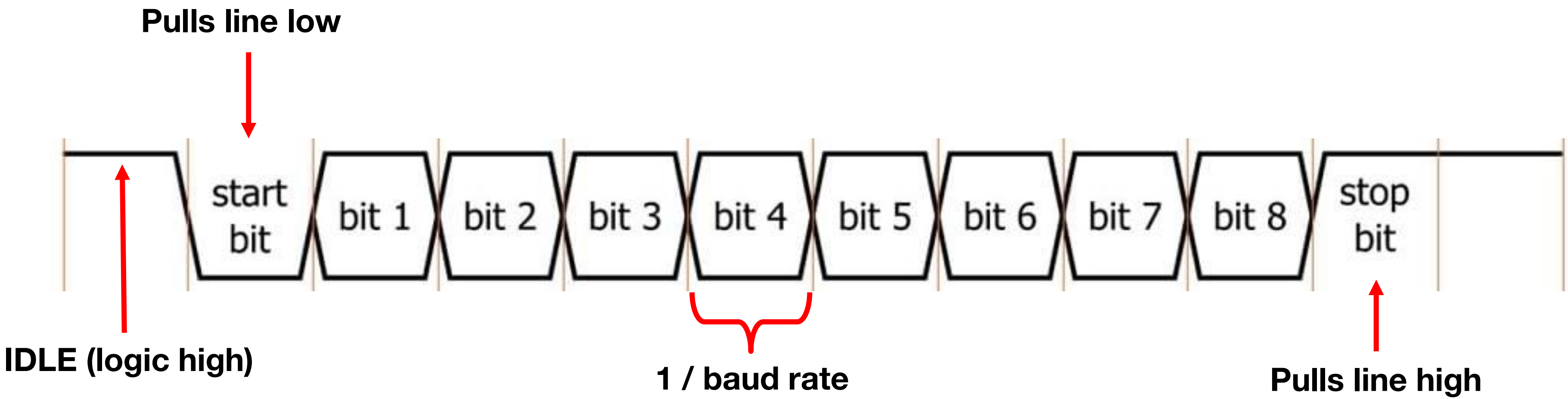

**Properties**

- **Asynchronous**
- **Limited to 2 devices**
- **Full-duplex**
- **Simple (only 2 communication lines)**

- **Tx: transmit**
- **Rx: receive**

# UART

## Timing Diagram




**Baud rate**

- User-controlled timing parameter
- Correlates with the time required to transmit 1 bit of data
- Common baud rates: 4800, 9600, 19200, 38400, 57600, 115200 bps

# UART

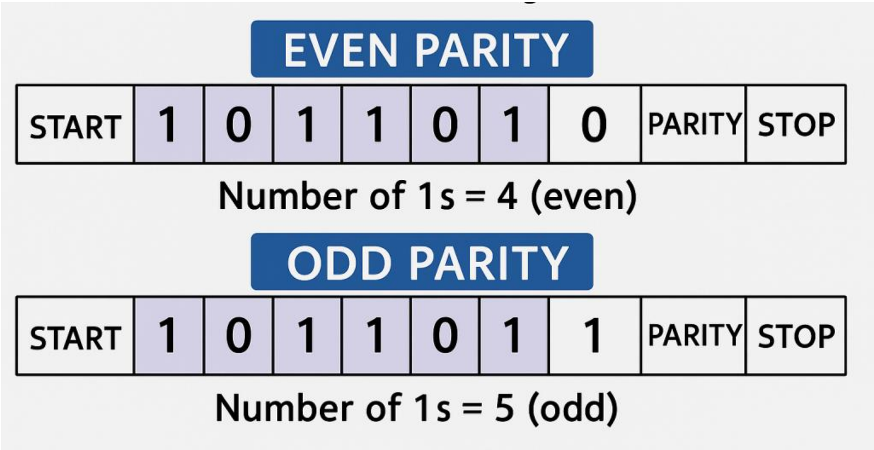
## Parity Bit

**How do we check for errors?**

**An extra bit can be used to match received data with a predetermined expectation of an even or odd number of 1s.**

**If the parity of the incoming data doesn't match the received parity bit, an error is raised.**

**Systems without parity bits often use two stop bits to prevent mistakes.**



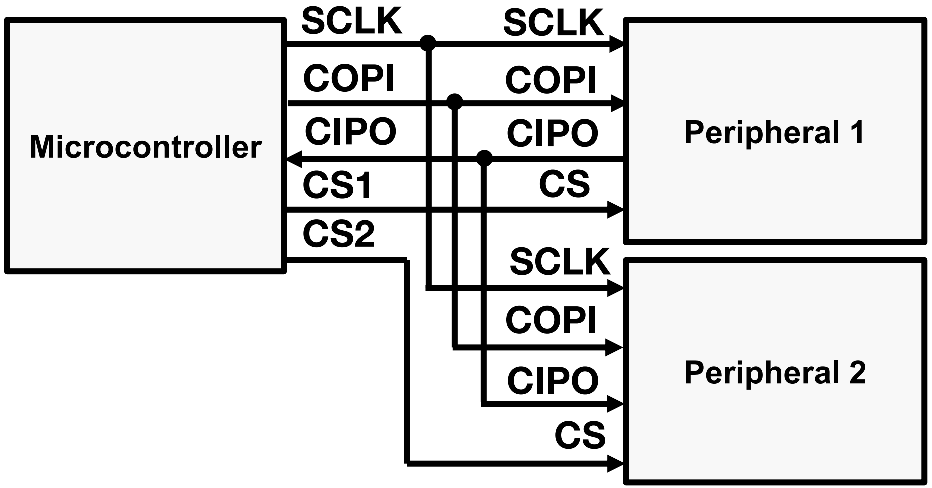
# SPI

## Overview



**What about if we want to use more than one device? Or we want to transmit data faster?**

**SPI (Serial Peripheral Interface)**




**Properties**

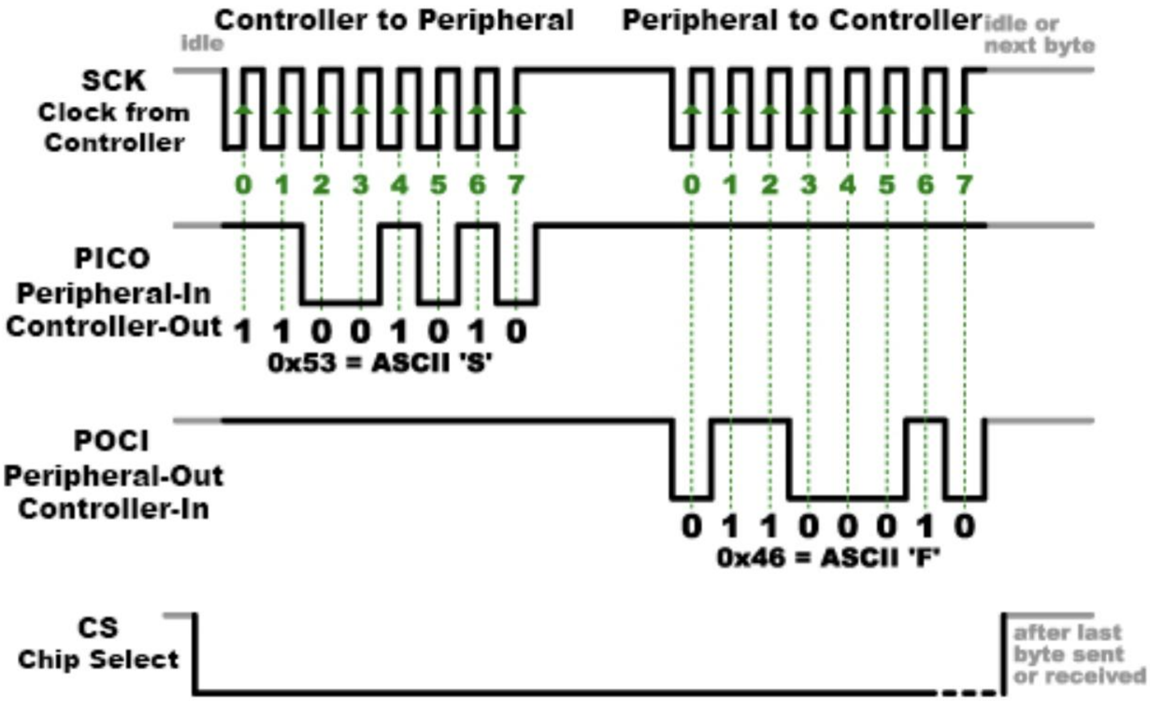
- **Synchronous**
- **Multiple devices**
- **Full-duplex**
- **4 communication lines**

- **SCLK: clock**
- **COPI: Controller Out Peripheral In**
- **CIPO: Controller In Peripheral Out**
- **CS: Chip Select**

# SPI

## Timing Diagram



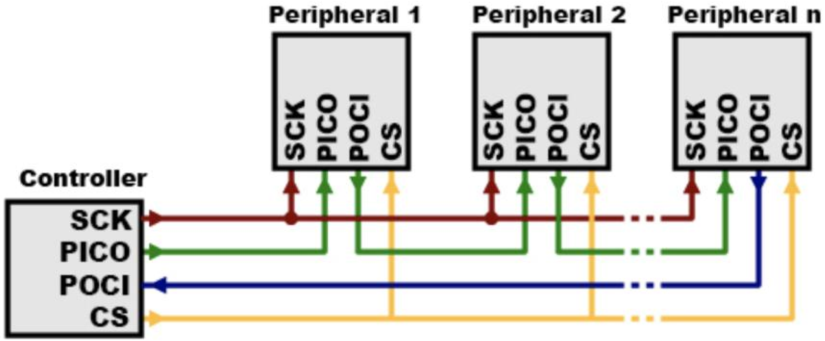


# SPI

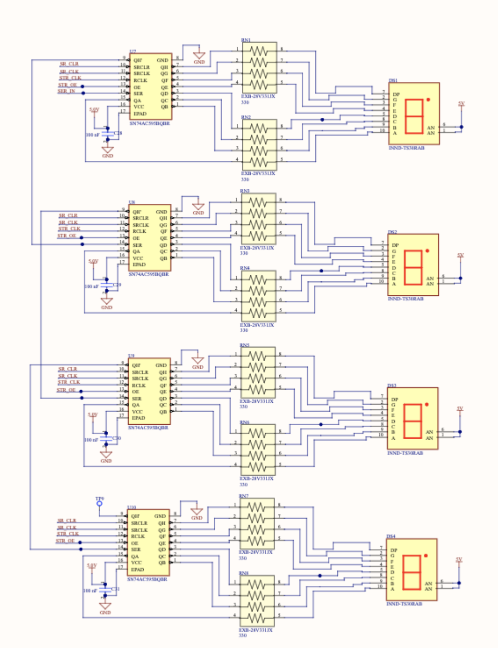
## Shift Registers

Data lines are daisy-chained together such that the Data Out of one device is the Data In of the next.

A single Chip Select line goes to every peripheral so that they are all activated simultaneously.




This is the same approach we used while driving our display!

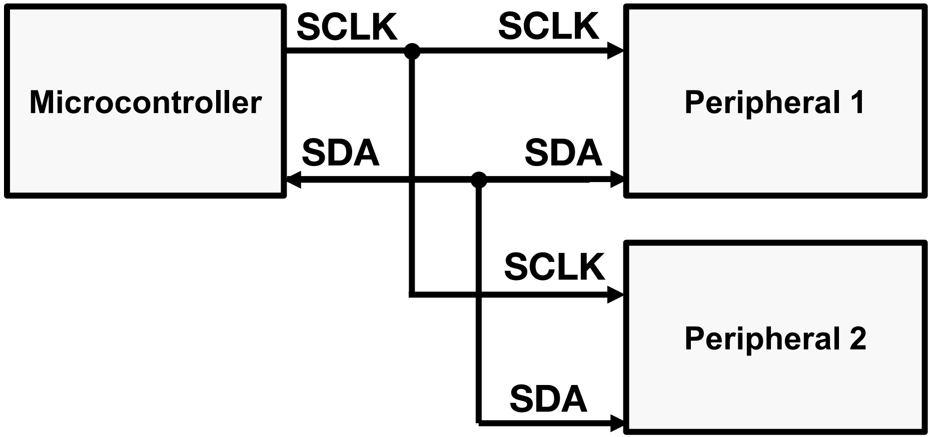


# I2C

## Overview

**Multiple devices is great…but we need to use more microcontroller pins every time we add one.**

**I2C (Inter-Integrated Circuit)**



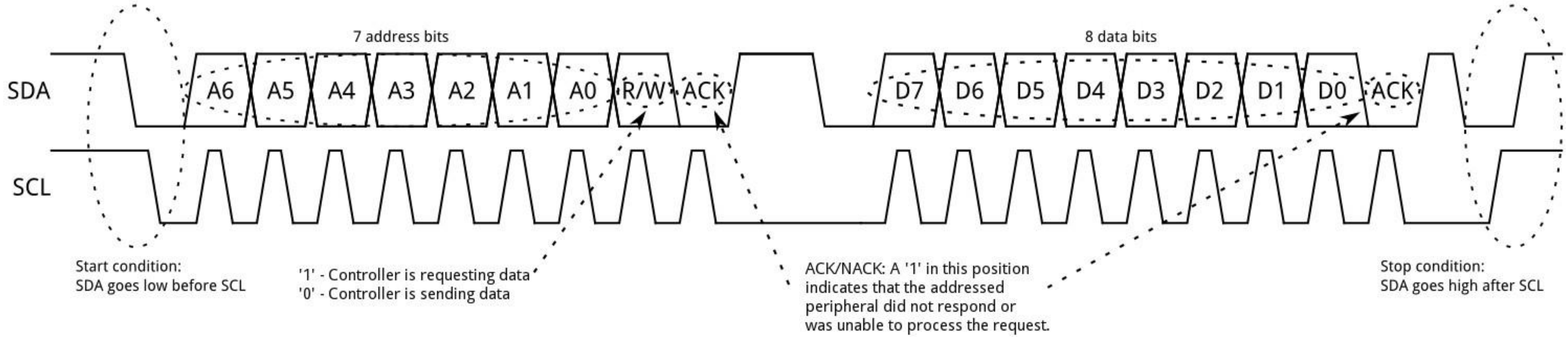

**Properties**
- **Synchronous**
- **Multiple devices**
- **Half-duplex**
- **2 communication lines**

- **SCLK: clock**
- **SDA: data**

# I2C

## Timing Diagram

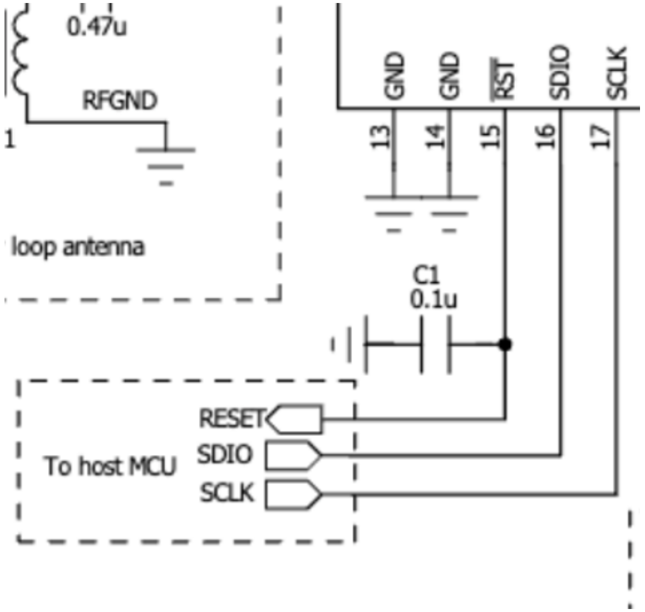





- **Address frame to select peripheral**
- **Start bit: Controller pulls SDA low while SCL high**
- **Acknowledgement bit (NACK/ACK): Peripheral pulls SDA low**
- **Data frames to pass data between controller and peripheral**

# I2C

## FM Receiver Chip



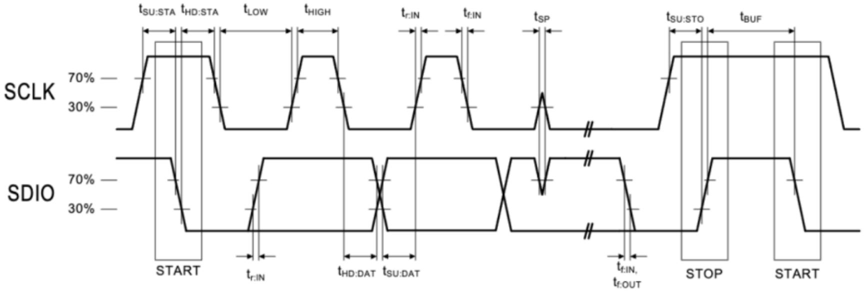




Figure 2. 2-Wire Control Interface Read and Write Timing Parameters

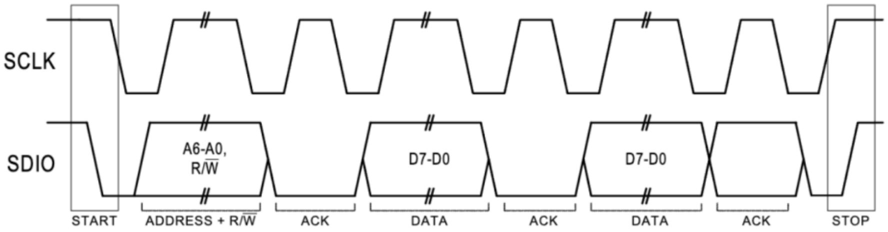



Figure 3. 2-Wire Control Interface Read and Write Timing Diagram

# CAN

## Overview

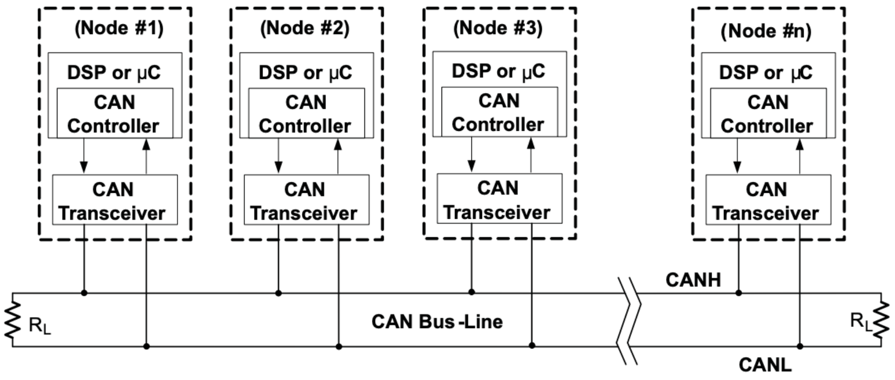
**SPI and I2C devolve across long distances because of noise/EMI/increased wire capacitances.**

**If the reference clock breaks, the transactions become unreliable.**

**Solution: Differential pair instead of single-ended ground reference**

**CAN: Controller Area Network**

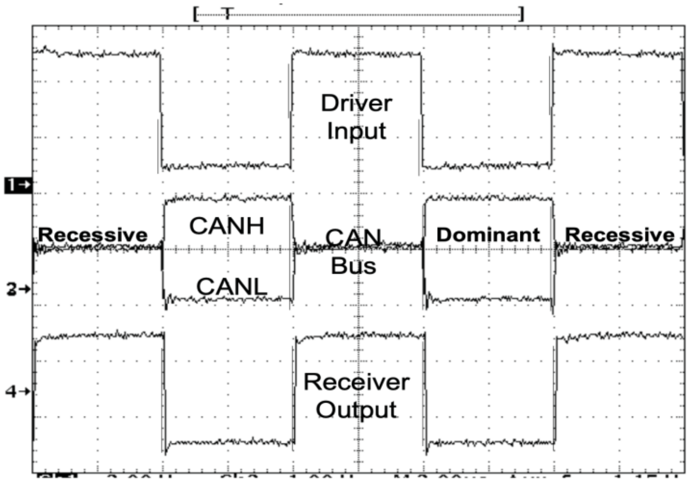
- **Widely used in automotive and industrial applications**
- **Network split into ECUs (Electronic Control Networks)**
- **Asynchronous – every ECU on a CAN network agrees on a bit rate**



# CAN

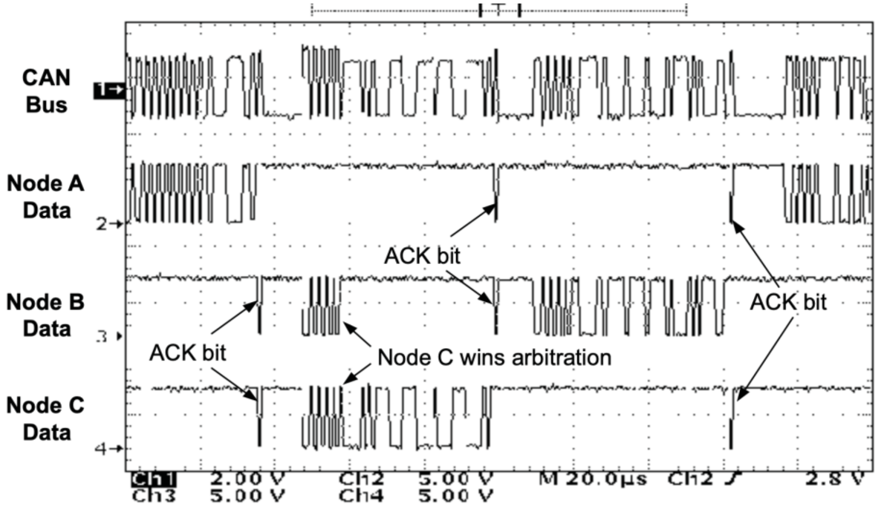
## Communication

**Dominant and recessive states used to communicate priority.**




**CANH and CANL equal in recessive state and pulled apart in dominant state.**

**Dominant bits overtake recessive bits during arbitration.**



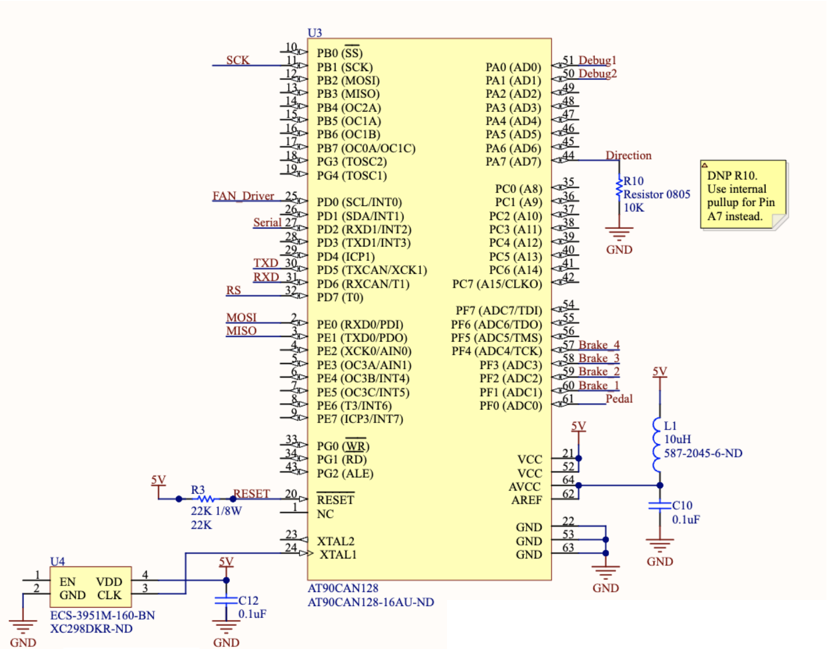

**Node A sends data to Node B and C. Node C produces a dominant bit first and sends data. Then Node B sends data.**

# CAN

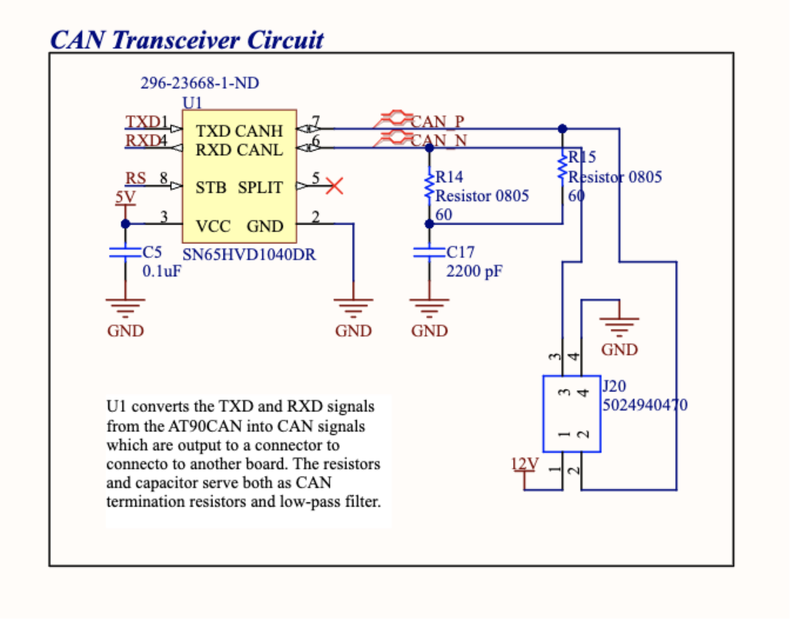
## Solar Electric Vehicle Team

THE ART AND SCIENCE OF PCB DESIGN

IAP 2026




**AT90CAN controller used in every ECU on the car.**




**AT90CAN TX and RX signals are converted into a CAN_P and CAN_N differential pair.**

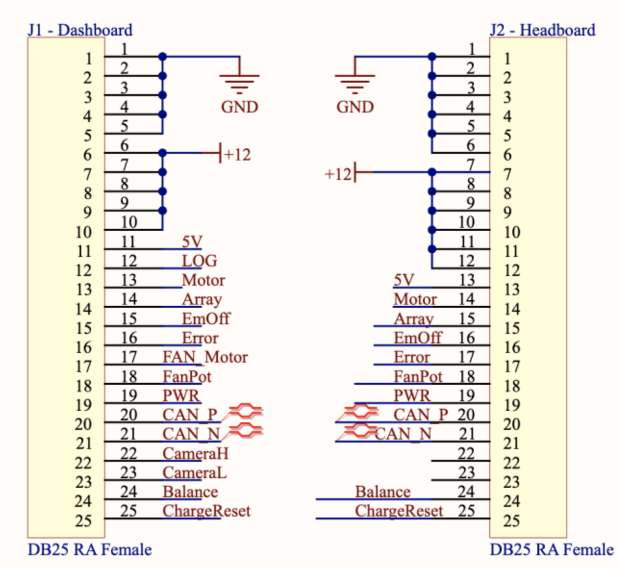
@ MIT

THE ART AND SCIENCE of PCB DESIGN
IAP 2026

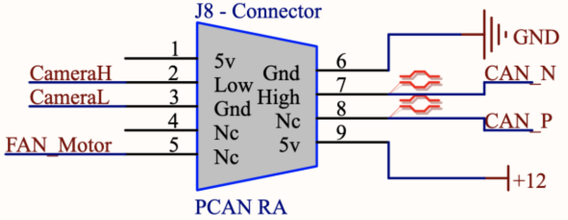
# CAN

## Solar Electric Vehicle Team (cont.)

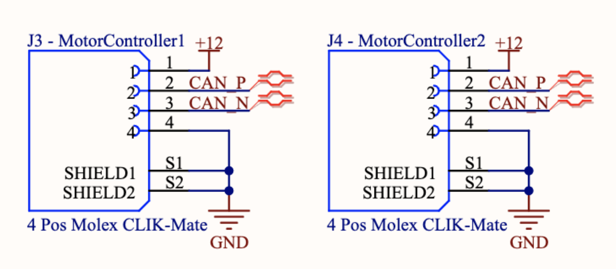
**CAN_P and CAN_N are used to communicate between parts of the car.**



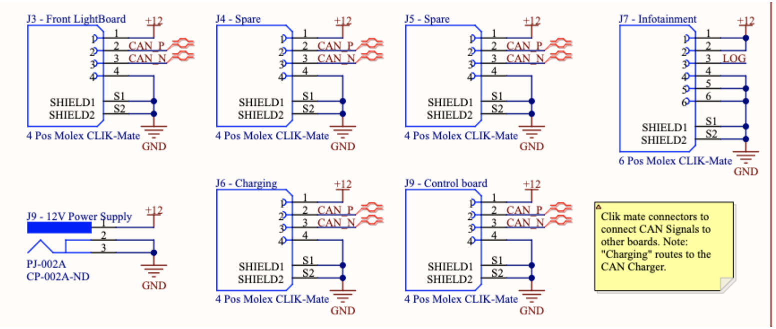

Dashboard and headboard




Backup camera signals




Motor controllers




Lights, infotainment, etc.



@MIT

# Conclusion



**There are many ways for parts of our system to communicate.**

**Communication protocols vary on:**

- **Form of timing**
- **Number of devices involved**
- **Communication speed**
- **Addressing mechanism**
- **Error avoidance mechanisms**
- **Number of communication lines**

**Between our analog design and our digital design, we have covered all the essentials of our physical system!**

# Final Notes

**Finish your boards!**
- **Office hours 1pm-5pm today**

**No recitation or lecture quiz**

**Fill out course evaluations by 5pm**

**Thank you all for a wonderful IAP!**