system need considering how to read analog data

add the GPIO pin no. for some codes

1. A4950 line 23
2. HX711\_V2 line 5
3. Sound line 10

rpi.gpio

rpi pwm for sound

**Checksum: The Error-Detection Mechanism**

**Transmitter Side (Sender)**

1. For every byte sent, the code adds its value to **\_chksum\_tx**:

cpp

\_chksum\_tx += val; *// Accumulate sum of all bytes*

1. After sending all data, it sends the **checksum byte**:

cpp

data\_buffering(byte(~\_chksum\_tx)); *// Send bitwise NOT of the sum*

* + The **~** (bitwise NOT) flips all bits (e.g., **0b10110011** → **0b01001100**).

**Receiver Side**

1. The receiver calculates its own checksum (**\_chksum\_rx**) while reading data.
2. After receiving all bytes, it checks:

cpp

if (\_buffer\_data[\_len - 1] == byte(~\_chksum\_rx)) {

*// Data is valid!*

}

* + If the last byte (received checksum) matches **~(sum of received bytes)**, the data is **valid**.
  + If not, the packet is **corrupted** and discarded.

**3. Why Bitwise NOT (~)?**

* Ensures the checksum doesn’t accidentally match corrupt data.
* Example:
  + **Sent data**: **0x01, 0x02**  
    Sum = **0x01 + 0x02 = 0x03**  
    Checksum = **~0x03 = 0xFC**  
    **Full packet**: **[0x01, 0x02, 0xFC]**
  + **If corrupted to [0x01, 0x03, 0xFC]**:  
    Receiver computes **0x01 + 0x03 = 0x04**  
    Expected checksum = **~0x04 = 0xFB**  
    But received **0xFC** → **Mismatch! Error detected.**