For homework 11, we will recreate the signal from our doorbell transmitter and send it out again using the HackRF. The series of steps below will make the process easier.

1. Replay Attack **[2 pts]**
   - For a bit of fun, we start with executing a replay attack. This indicates that our transmitter constantly sends the same signal (just capture a signal and play it again – any tool can be used)
   - To get the grade, use your phone or some other recording device to capture the replay attack
   - Upload the video to the link below. You should be able to see any other replay attack videos uploaded from other students there. My sample replay attack videos are also there :). Remember to name you video with an identifier linked to you (e.g. your name, student code)
     - Replay Attacks
     - ***Password***: wire_tech_sec_autumn_2023

2. Find the Bit pattern **[1.5 pts]**
   - Write down (type :)) the bit representation for a symbol 0 and a symbol 1 (i.e. 77/25 or 66/33 PWM)
   - Write down the full bit representation of the signal (i.e. if 4 bits used per symbol and there are 25 symbols then 100 bits should be written)
   - Screenshots are welcomed – showing the length of symbol etc...

3. Find the Symbol Duration **[0.5 pt]**
   - Find the length of time needed to transmit one symbol
   - Take a screenshot of the tool used to find this value *(as shown in the lab manual)*
   - Write down any calculations performed (i.e. if Audacity is used)

4. Find the Bit Duration **[0.5 pt]**
   - **Calculate** the length of time needed to send one bit using the symbol duration value from 3.

5. Find the Pause Duration **[0.5 pt]**
   - Find the time elapsed between each sent packet *(the pause time after each symbol pattern)*
   - Take a screenshot of the tool used to find this value *(as done in the lab manual)*
   - Write down any calculations performed (i.e. if Audacity is used)

6. Use the GNU template script provided to send the bit pattern **[5 pts]**
   - Add a python module to the homework template and create a function that can generate the bits needed to simulate this doorbell pattern **[3 pts]**
     - The function should convert the symbol pattern to bits
     - And add the appropriate bit pattern to simulate the pause between packets
     - Use this function to set the value for the `Vector Source` block
   - Set `operation_freq` to the frequency observed in Homework 10 **[1 pt]**
     - Use this variable to set the frequency for the `HackRF` and `Signal Source` block
       - The HackRF transmits the bits on this frequency
       - The Signal Source modulates the bits on this frequency
   - Set the `symbol_duration` variable to the value calculated in Step 3 **[0.25 pt]**
   - Set the number of bits that represent each symbol in the `bits_per_symbol` variable **[0.25 pt]**
   - Calculate the `bit_duration` value using the `symbol_duration` variable and `bits_per_symbol` variable (i.e. add the equation needed for the calculation ***not*** the calculated value) **[0.5 pt]**
   - Enable the `File Sink` block (*larger file*) or the `Wav File Sink` block (*and Complex To Float*)
     - Save a file of the transmitted signal.
     - **Compress and upload this file to Moodle**
   - Optional changes

- The sample rate, though 2Msps should be adequate to complete the task (using GNU Radio 1Msps may also be possible)
- The IF Gain on the HackRF to improve the range of the transmitted signal

Hint: If the doorbell does not ring, check the length of time the symbol bits are sent. Try capturing the original signal with a sample rate of 1Msps and see if this matches the timing of your recreated signal. That is, compare the file generated from 6 with the 'real' signal. Using the .wav file of the generated signal should show the true length of time the bits are sent. Please also read the "Confirming Bit Duration and Timings" section in the Lab manual.

Add-ons: Use the GUI Toggle Button (available in GNU Radio 3.10) to simulate the doorbell push button transmitter. When the button is pressed the doorbell signal is sent hence ringing the doorbell. When the button is released the signal is not sent hence the doorbell remains silent.

Note: Please email me if an uploaded video needs to be removed.
Note 2: Please be careful of the things recorded in the video *i.e.* do not record something you do not want others to see/hear.

Due Date: Check Moodle for date
Late Submission Deadline: Check Moodle for date (usually 1 week after due date)
Submission Files:

- **Location**: Moodle Homework 11 link
  1. a file with the answers and screenshots for steps 2 to 5
  2. the modified GNU script template (and python module file) for step 6
  3. the file containing the generated signal from 6 (compressed e.g. `file.zip`)
- **Location**: Replay Attacks
  - video of your replay attack(s) for step 1
    - The link can also be accessed through Moodle