

Communication Protocol for Testing Impedance Microbiology

Synshine Engineering Team

Using the circuit provided, we will be scanning in the range from f-value closest to 30 Hz to 30 kHz. The increments are to be done in a factor of two. The circuit will be tested for 14 different frequencies.

The following things are black-boxed:

- `int init_adc()` : Initializes ADC. If it worked, then returns 0, else -1.
- `int init_sine()` : Initializes the sine wave generator. If it worked, then returns 0, else -1.
- `long read_val_adc(int a)` : This function does the following :
 - If 'a' is any one of 0,1,2, or 3, it returns the 24-bit value of the a^{th} ADC as a long .
 - If 'a' is not one of the above the function returns '-1'
- `int set_val(long freq)` : This sets the closest frequency possible in the sine wave generator. If it worked, then return 0, else -1.
- `int change_freq_steps(long n_steps)` : Adds $0001_{hex} \times n_steps$ to the output to the sine-wave generator. If it overflows, then return -1.
- `int mult_freq(long k)` : Multiplies the output to ADC by k. If it overflows, then return -1.
- `int set_sine(int a)` : If 'a' is 1, turn on the sine wave generator else turn it off.

The arduino responds to the command of the computer through serial communication.

1 Transmission Protocol for the Primary message

The primary message will be the initial message before validation procedures.

1.1 From Computer to Arduino

The computer can send the following :

- `SETFR:#` = This commands the arduino to set the frequency to the value closest to # which is a long
- `CGSTP:#` = This commands the arduino to change the frequency steps by # which is a long
- `MLSTP:#` = This commands the arduino to multiply the frequency by # which is a long
- `CHKCF` = This commands the arduino to check the current frequency
- `GENHI` = This commands the arduino to turn on the sine wave generator
- `GENLO` = This commands the arduino to turn off the sine wave generator
- `ERROR:<>` = This sends an error to the arduino

1.2 From arduino to computer

The arduino can send the following:

- `ERROR:<>` = This sends an error to the computer
- `SDDAT:#:c` = This sends the data to the computer as # = ADC output of the c^{th} ADC