EGB242 Assignment 2 (25%)

Orbiting the Red Planet

Semester 1 2022

Released: Friday, 13th of May at 11:59pm
Group Plan Due: Friday, 20th of May at 11:59pm
Final Assignment Due: Friday, 3rd of June at 11:59pm

Message from MARS-242 Mission Control

The chief engineers at BASA Headquarters are confident in your demonstrated skills dealing with the simulated Mars mission scenarios you have worked on so far. You are now ready to be part of the main engineering team supervising the 'live' MARS-242 mission.

Ensuring communications are stable and uninterrupted are critical to the mission's safety and success. As the spaceship is orbiting Mars, identifying an appropriate landing site for the landing module is of high priority. Fuel reserves on-board the spaceship and the effects of planetary alignment require that this process be carried out efficiently and precisely.

Once the crew is safely deployed on the planet, it will be your job to ensure the recovery of vital acoustic information which will be used for potentially improving the safety of this and future crews.

You are now called upon to assist with achieving these two high priority objectives. Your contributions will strongly depend on the techniques you have practiced in the simulated tasks code-named Assignments 1A and 1B.

Section 0 - Preparation

Read the entire document before attempting the tasks.

There is important information at the end of this assignment brief regarding how the assignment is to be presented. A unique set of data is provided to each group based on the student numbers of your group members. You will need to generate this data in order to complete the assignment.

- Extract the files from `EGB242_Assignment2.zip`
- Open and read `GenerateAssignment2Data.m`
- Enter your group members' student numbers in the appropriate location.
- If your group has fewer than 3 members, enter 0 for the missing member(s).
- Ensure all necessary files are in the same folder and run the script.
- Two (2) files will be generated that you will use for this assignment - `A2P2Data.mat` and `A2P3Data.mat`.

Two (2) additional template files, `A2part2.m` and `A2part3.m`, have been provided to you, corresponding to sections 2 and 3. You are to write all code for this assignment for each section in the respective file.

Section 1 - Mars-242: Mars Rover

The maths carried out for this section can be either typeset or handwritten neatly and scanned into the report.

In anticipation of landing on Mars soon, a rover has been gathering and collecting important data on an appropriate landing site for the mission. The rover has been communicating directly with stations situated on Earth where BASA Engineers conduct their analysis on the received data and send the results to the astronauts for action.

Unfortunately, a malfunction with the communication system on the rover has meant that this very long distance communication is no longer possible. Because the rover's data consists of information and images of the planet, which are vital for the success of the anticipated Mars landing, a solution has to be found. The best alternative is for the rover to send the data directly to the spaceship and for the astronauts to do the analysis themselves.

However, this can only succeed with your help and knowledge of how Linear Time Invariant systems work. You are called upon to assist the astronauts to mathematically model a system that allows receiving and processing of the rover data.

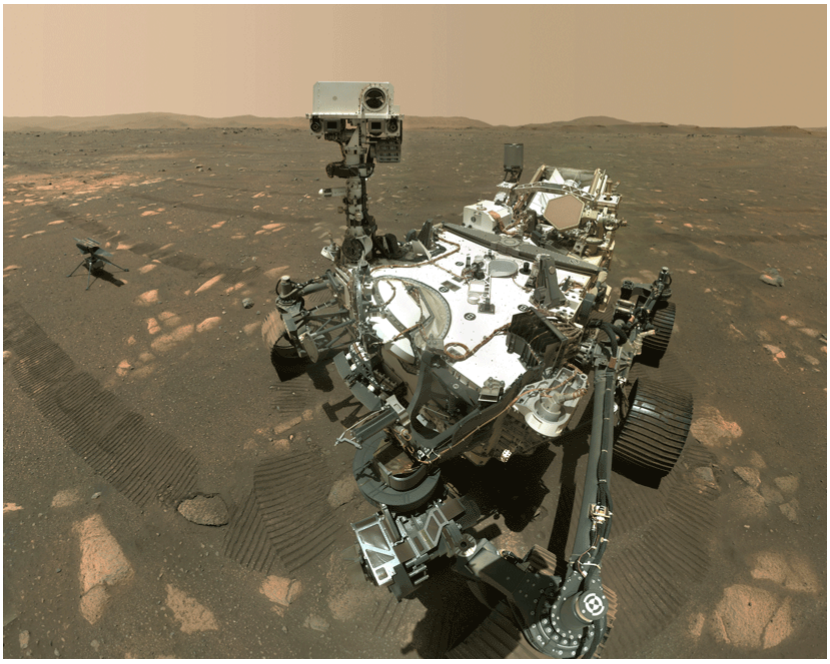


Figure 1: The Mars 2020 *Perseverance* rover (Source: NASA).

> *Your task is to mathematically determine how communication between the spaceship and the rover can occur.*

1.1 The spectrum of the rover's transmitted signals may interfere with the spectrum of the spaceship's own communication module. The spaceship's communication channel can be modelled as a constant magnitude function in the frequency domain, between $-f_{bw}$ and $+f_{bw}$. The f_{bw} for your group will be printed out when you generate your data. Mathematically express and sketch this function. (CR 1a)

1.2 The rover's transmission function can be expressed as:

$$s_{\text{rov}}(t) = A\text{sinc}^2(4000t)\cos(2\pi \times 8000t)$$

The magnitude variable A is printed when you generate your data. Derive an expression for the corresponding representation of this function in the frequency domain, $S_{\text{rov}}(f)$, and sketch the magnitude spectrum. (CR 1a, 1c)

1.3 Determine the transfer function of the ideal filter, $H_{\text{filt}}(f)$, that is required to eliminate the spaceship's broadcast information and ensure the rover's signal is received fully. Ensure that **all** of the interference is removed as any overlap in the spectrum's may result in corrupted data. Additionally, name the filter type. Ensure that you select the simplest type of filter for this situation and justify your selection. (CR 1a, 1d)

1.4 Using your knowledge from Assignment 1B, recover the frequency shifted rover's signal graphically from the filtered signal and then design an ideal filter to remove the high frequency copies. (CR 1a, 1b, 1c)

1.5 Evaluate and sketch the final frequency domain function that defines the range of frequencies you will receive from the rover after applying your ideal filter from above. Is any information lost due to the application this filter? Justify your answer to this using you knowledge of the Fourier transform. (CR 1a,1d)

1.6 Discuss what practical impacts applying the filter could have on receiving the rover's information. Explain how you might mitigate these issues and the considerations which you should take into account when applying this filter. (CR 1d)

Section 2 - Mars-242: Choosing a landing site

All code for section 2 should be entered into the file A2part2.m. Make sure to include appropriate scaling and informative axis labels on your plots.

The Mars-242 crew must now identify a suitable site for their landing module. The surface-based rover has transmitted images of some areas it has scouted to the spaceship for human inspection. After all your efforts to enable communication between the rover and the Spaceship (in section 1: Mars Rover), you now discover that the transmission channel between the rover and the spaceship has introduced both periodic and bandlimited random noise to the signal stream. As you can see from the example image shown in Fig. 2, the noise has rendered the image useless for identifying any surface features.

Your task is to de-noise the images and recommend an appropriate landing site.

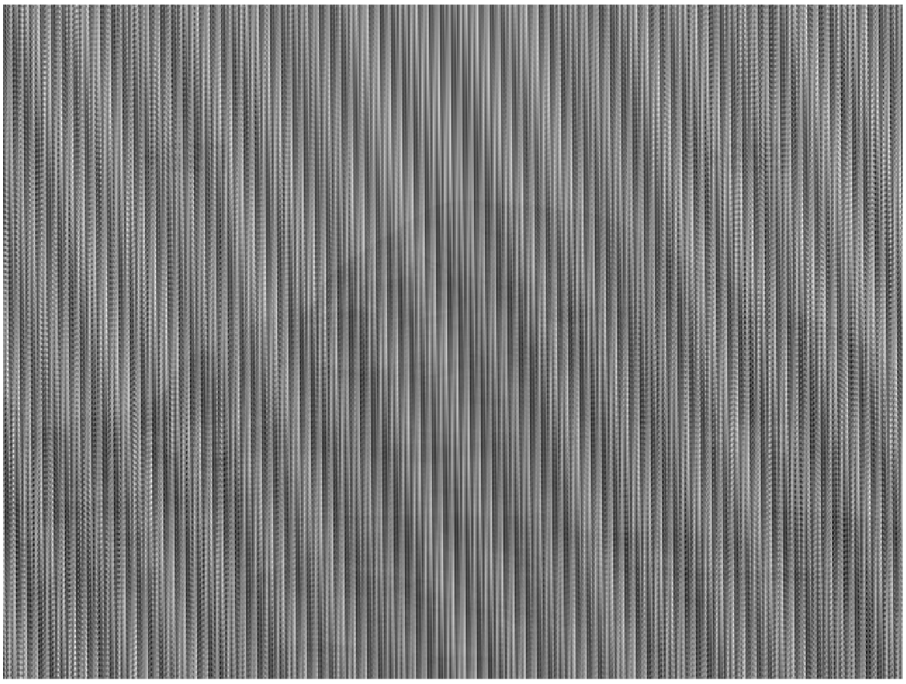


Figure 2: A corrupted image from the Mars exploration rover.

Images in MATLAB are stored as 2D matrices. Each element in the matrix represents a pixel of the image. The numeric value at each matrix index describes the colour intensity of each pixel. Matrix elements for grayscale images are floating point numbers between 0 and 1; which correspond to the colours black (zero intensity) and white (maximum intensity) respectively. You will be working with grayscale images in this assignment.

Even though the images are represented by 2D matrices, they are received as a 1D data stream. The first received pixel is placed at the top left corner of the image. Subsequent pixels are used to fill the image column-wise (top to bottom) from left to right *(shown below)*. The size of the images being received are 480×640 pixels.

$$1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad 9 \quad 10 \quad 11 \quad 12 \longrightarrow \begin{matrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{matrix}$$

The received signals are provided in the rows of the matrix ***sig***. There are 4 images received, and hence 4 signals, with the first stored in the first row of the matrix.

NOTE: For sections 2.1-2.10 only use the first received image.

2.1 *View the noisy image* - Display the first image using the `imshow` function. You will need to first convert the signal into a 2D matrix using `reshape`, *ie.*, `imshow(reshape(sig(1,:), 480, 640))`. (CR 2a)

2.2 *Reference vectors* - Image data is received from the rover at 1000 pixels/sec. Consider each pixel a single sample. Construct a time vector for the input signal (i.e., a single image) and store this in ***t***. Create an appropriate frequency vector and store in ***f***. (CR 2a)

2.3 *Visualise the received signal* - Plot the first received signal (***sig(1,:)***) in both the time and frequency domains. Scale the time domain plot to show the first 3 seconds of the signal. Designate the frequency domain data with capitals i.e ***SIG***. Data in ***SIG*** should correspond with time domain data in ***sig*** i.e., row-wise. Identify (graphically) both the periodic and bandlimited random noise in both the time and frequency domain plots. Ensure you state the approximate period of the periodic noise and the bandwidth of the bandlimited noise. (Rescale the plots if necessary). (CR 1b, 2b)

The periodic noise in the image can be removed if it is accurately estimated. Engineers at mission control have identified a set of possible values for the period of the corrupting noise. These values are stored in seconds in the vector ***candidateT***. Averaging the shape of the waveform for all periods will emphasise the noise component, smoothing away the underlying image into an offset value. The function `estimateNoise.p` has been provided to you to perform this noise identification. It will return a vector one period long representing your periodic noise.

2.4 *Estimate the periodic noise* - Determine the period of the noise from the given options and store your selected value as ***T***. The units for ***T*** should be *seconds*. Use the provided `estimateNoise` function to estimate the noise profile and store the result in the variable ***Noisesig***. The syntax for its usage is

```
Noisesig = estimateNoise(inputSignal,periodInSamples)
```

Plot and compare a periodic version ***Noisesig*** to the received signal (***sig(1,:)***), showing only the first 3 seconds. You can use the MATLAB function `repmat` to repeat a vector many times. *If the overall shape of the signals do not closely match, you may have an incorrect estimate of the period.* (CR 1a, 1c)

2.5 *Model the periodic noise* - Model the noise signal by computing the Fourier coefficients of ***Noisesig***. You can choose to perform this using either the trigonometric or complex exponential Fourier series. Start by computing the DC (a_0) and the first 6 harmonics (a_n, b_n for $1 \leq n \leq 6$ or C_n for $-6 \leq n \leq 6$). Remember to account for sampling rate. List these coefficients and explain your process for computing them. (CR 1a, 1b, 2a, 2b)

2.6 *Bias* - Mission control has determined that the mean of the periodic noise signal is 0. Any DC component present in ***Noisesig*** is due to the underlying image. Make any changes necessary to your coefficients to take this information into account. List the new coefficients and explain the reasoning behind any modifications you have made. (CR 1a, 1d)

2.7 *Generate the approximation* - Using these Fourier Series coefficients, generate the approximation of the noise component to correspond with the received signal (ie. all 307,200 samples) and store it in ***Noisesig_fs***. (CR 1a, 2b)

2.8 *Compare the approximation* - Compare the first T samples of ***Noisesig_fs*** to ***Noisesig***. Does the number of coefficients used result in a good representation of ***Noisesig***? (CR 2b)

2.9 *De-noise* - Use your previous BASA training from Assignment 1A to de-noise the image. Store the result in the first row of an image matrix ***im1***. Display the image and its spectrum in MATLAB. Include the recovered image and spectrum in your report. Comment on how the spectrum has changed compared with the spectrum of the received image. Also comment on the quality of the de-noised image. Has a sufficient amount of noise been removed to identify details of the image? If not, experiment with using more Fourier coefficients to improve the image quality. Justify the number of coefficients you decide on using for removing the noise. (CR 1c, 2b)

2.10 *Remove the bandlimited random noise* - Remove the bandlimited random noise. Decide whether to do this in the time or frequency domain, but the final outcome must be in the time domain. Store the filtered signal in the first row of ***im2***. (CR 1d)

2.11 *Choose a site* - Mission control has determined the periodic noise profile is consistent across all received data. Using ***Noisesig_fs***, repeat the de-noising process from 2.9 for the remaining 3 images and store in the corresponding rows of ***im1***. View the spectrum and remove the bandlimited random noise using the same method used in 2.10. Store the results in corresponding rows of ***im2***. Display the images contained in ***im2*** in a single figure in MATLAB. Of the 4 landing sites photographed, which is an appropriate site to send our lander? Justify your choice. (CR 2b)

2.12 *Resolution* - Navigational numbers were marked onto each image at the time they were photographed. Identify the numbers in the images and list these in your report. Also include the filtered images in your report, at a scale that allows the navigational numbers to be read. (CR 3d)

> MATLAB variables that should be included in your workspace for section 2 (A2part2.m),
>
> **t** - Time domain vector
> **f** - Frequency vector
> **SIG** - Fourier Transform
> **T** - Selected candidate T value
> **Noisesig** - Estimated noise signal
> **c0, cn** - Complex Fourier series variables, OR, **a0, an, bn** - Trig Fourier series variables
> **Noisesig_fs** - Approximation of noise
> **im1** - Image matrix 1
> **im2** - Image matrix 2

Section 3 - Mars-242: Habitat impulse analysis

All code for section 3 should be entered into the file A2part3.m. Make sure to include appropriate scaling and informative axis labels on your plots.

After successfully choosing a safe landing site for the astronauts, a base of operations has been set up on the surface out of a single large portable habitat structure. For safety purposes it has been decided that additional methods of determining crew member locations inside the habitat is vital for quickly responding to emergencies.

One such method that is being investigated is the modelling of the acoustic response of different rooms inside the habitat. Knowing the acoustic response of a specific room may allow for the identification of crew member locations in an emergency when they cannot themselves indicate where they are. As a result, the team has individually taken impulse responses of different rooms inside the habitat and has transmitted them back to BASA headquarters.

The impulse response signals and a text message indicating which crew member took the measurement and the room they were in, have been multiplexed using the Fourier Transform modulation property. The result is a Frequency Division Multiplexed (FDM) data stream that has been transmitted to mission command. By using the modulation property, a signal can be shifted to occupy a new band of frequencies. If several signals are modulated onto carrier frequencies far enough apart such that their spectra do not overlap, multiple signals can be contained and transmitted in just one signal (multiplexing).

Your task is to extract the room impulse response signals and the text of each individual room, remove any noise or interference effects from them, tabulate the room name and response and apply some of the impulse responses to your our audio and comment of the results.

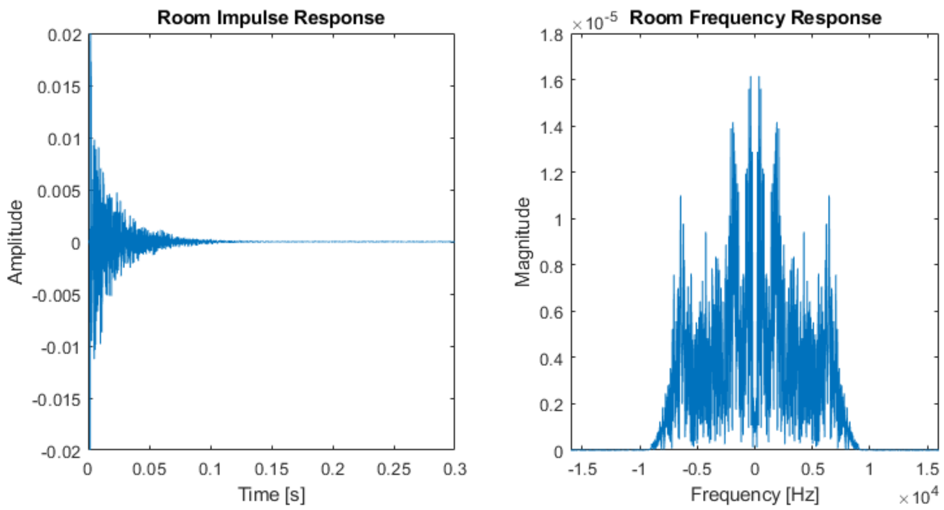


Figure 3: Room impulse response signal - (*Left*: Time domain, *Right*: Frequency domain)

The impulse and frequency response of an unknown room is shown in Fig. 3. In the time domain, this impulse response attenuates away quickly however this will not always be the case for all of the responses. As a result, all of the impulse response signals you will receive are 0.5 sec long. In the frequency domain, all of the energy will be contained within a ±8kHz bandwidth due to the recording of the impulse. Therefore, we can say that the bandwidth of the impulse response is $B = 8\text{kHz}$.

The multiplexed signal is normally received at command centre by a modular hardware system which consists of a receiver (with real-time spectrum analyzer) followed by a frequency shifting module and an analogue filter. The output of this filter is sampled by an analogue-to-digital converter for further manipulation. The block diagram of the system is shown below in Fig. 4.

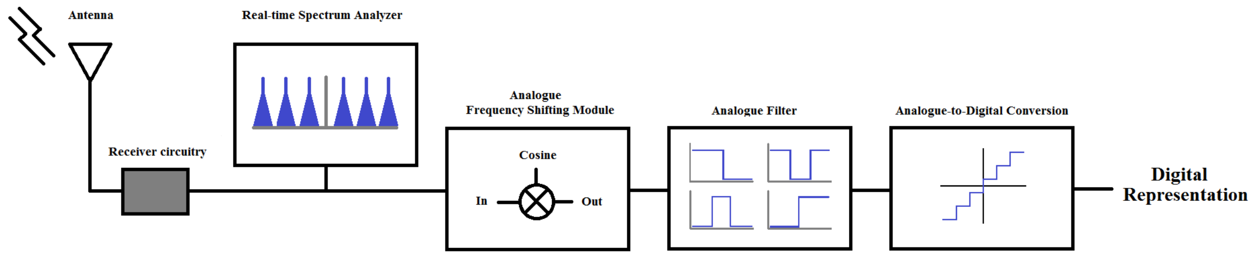


Figure 4: Block diagram of the modular receiver

3.1 *Represent the spectrum analyzer* - Determine the multiplexed signal's sampling period, `Ts`, and construct a time domain vector, `t`, that corresponds to `muxSignal`. Plot `muxSignal` against this time vector. Compute the Fourier transform of `muxSignal`, `MUXSIG`, and construct a suitable frequency vector, `f`. Plot the magnitude spectrum of `MUXSIG`; there should be 6 clear peaks indicating the multiplexed signals. (CR 1a, 2a)

3.2 *Determine demultiplexing parameters* - Identify the frequency shifts and store them (in an ascending order) in the row vector `fshift`. Find the corresponding magnitude and phase and store them in the row vectors `Mag` and `Phase`, respectively. *Note:* Remember to scale `Mag` to account for the sampling rate. (CR 1b, 2b)

3.3 *Remove the frequency shifts for all 6 signals* - Input variables `muxSignal`, `t`, `Mag`, `fshift`, and `Phase` should be inserted into the function `FDMDemux` as row vectors (see below). Each row of the `xdm` matrix will contain either the room impulse response, or text data corresponding to a specific room's impulse response with frequency shift removed upon successful application of the module. (CR 1b, 1d)

The frequency shifting module is provided in `FDMDemux.p`. This is mathematically equivalent to the hardware module being used, which demodulates a modulated signal in a similar manner to Assignment 1B. The function additionally ensures, using the carrier's magnitude and phase, that the originally transmitted signal is correctly restored.

The syntax for its usage is:

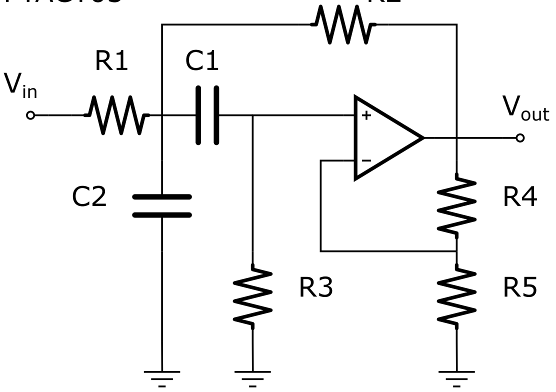
```
xdm = FDMDemux(muxSignal,t,Mag,fshift,Phase)
```

3.4 *Review* - Compute the Fourier transform for each data stream in `xdm`. Store the result in the matrix `XDM`. Plot the magnitude spectrum for each data stream and comment on the results. (CR 1a, 2a, 2b)

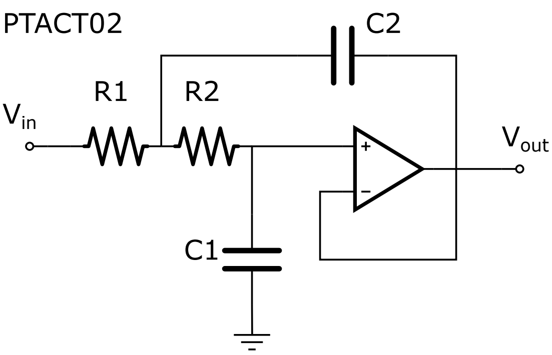
With the frequency shifts successfully removed using the `FDMDemux` module, your team was about to move onto filtering away the high frequency copies left behind due to the removal of the frequency shifts when the current filter module became non-operational. Your expertise in using the Laplace Transform for circuit analysis will be required to select and implement an appropriate replacement filter. Circuit diagrams from the unit's technical documents have been supplied below along with their corresponding transfer functions. You must analyse the response of the circuits to determine which filter is the most appropriate for this application. Once your signal has been filtered, you should remove any additional unwanted frequency components and proceed with your analysis.

You have been given the four most appropriate filters out of the six shown below. The transfer functions for these four systems are stored in ***sys***, with the corresponding circuit diagram reference in ***sysInfo***. Each row of the matrix is the description for each of the systems. To display the part name and parameters for the first system, type ***sysInfo(1,:)*** .

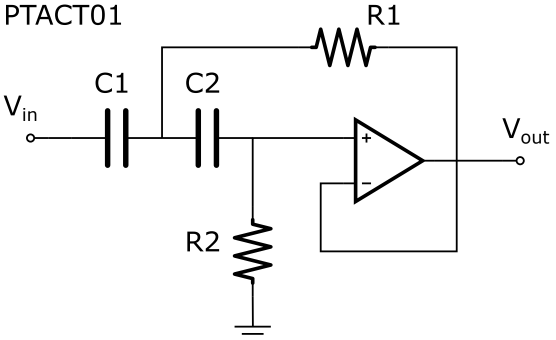
PTACT03



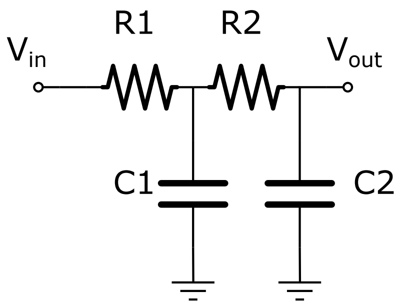
PTACT02



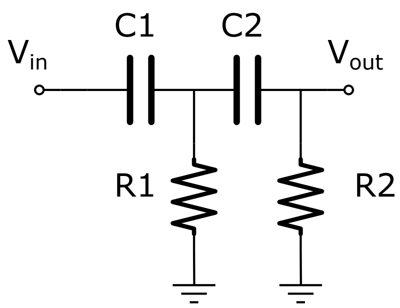
PTACT01



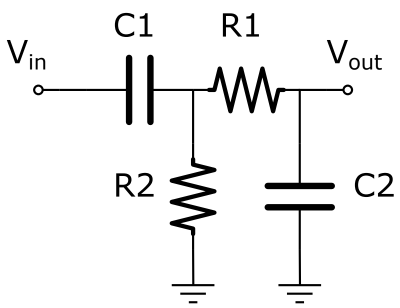
PTPSV03



PTPSV02



PTPSV01



3.5 *Mathematical analysis* - Display each transfer function in MATLAB. Determine the suitability of each system (for the task) based solely on its denominator polynomial. Justify your choices. The function `factorTF.p` has been provided to perform factorisation of the polynomials if needed. Use it by passing a transfer function to it, *eg.* `factorTF(sys(1))`. (CR 1a, 1b, 1d)

3.6 *System Analysis* - Analyse each system to help you understand their behaviour. You may use the LTIviewer or other functionality in MATLAB to find information such as the: impulse response, step response, bode plot and pole zero map. (CR 1a, 1c, 1d, 2b)

3.7 *Recommend* - Make your final filter selection and provide a short summary (between 50 and 100 words) justifying your choice based on your mathematical and system analysis performed in Section 3.6 above. Describe any implications that may come from using the chosen type of filter. (**As an engineer you should be able to explain your decisions qualitatively and quantitatively**) (CR 1d, 2b)

3.8 *Filter the signals* - Use the system you have selected to extract the room impulse responses and the text streams. The same filter parameters can be used for each signal. Do this by first multiplying the transfer function of the system by each data stream in the frequency domain (`XDM`). Store the result in the matrix `MSG`.
Convert these back to the time domain and store the result in a matrix called `msg`, remove any DC component that still remains due to the filtering process and scale the values in vectors between -1 and 1.
Each row of both `msg` and `MSG` should contain the time domain and frequency domain of either a room impulse response or text stream, respectively. Organise the rows so that they correspond to each frequency in `fshift`. Plot all of the signals in both the time and frequency domains. Ensure that plot axis values have been chosen to ensure the signals can be easily interpreted. (CR 1b, 1c, 1d)

3.9 *Extract and resample impulses* - Extract the room impulse responses (rows 1,3,5) from `msg` and using the MATLAB function `resample`, resample all of the impulse responses to `FS_recov` = 16kHz and store them in a variable called `impulses_recov`. (CR 1a, 2a)

```
impulses_recov = resample(msg(1:2:end,:).', FS_recov, FS).'
```

3.10 *Extract text messages* - Extract the text messages (rows 2,4,6) from `msg` and store the results in a variable called `texts_recov`. (CR 1a, 2a)

3.11 *Decode and plot* - Produce figures of each room's impulse and frequency response with the room name and which astronaut took the reading. *Note:* You will need to construct a new time and frequency vector for plotting the resampled room impulse and frequency responses using a sampling rate of `FS_recov` = 16kHz. Call the vectors `t_recov` and `f_recov`. (CR 2a, 3c)

This module takes a single text stream as input and returns the decoded char array message. Call this function in a loop, passing in each text stream individually.

The syntax for its usage is:

```
[text] = decoder(msg)
```

3.12 *Practical room impulse analysis* - Record your own voice for approximately 3sec and using the MATLAB function `audioread`, load your audio into the workspace and resample your audio to `FS_recov` = 16kHz using the MATLAB function `resample`.
Apply the impulse responses to your recorded audio in either the the time or frequency domain and listen to the results. Comment on any changes that you hear and if necessary show figures depicting your audio before and after the application of the room impulse

responses. **Important:** You will need to extend the length of the impulse responses from *impulses_recov* using zeros to match that of your recorded audio vector if you wish to perform this task in the frequency domain. (CR 1a, 1b, 2a)

MATLAB variables that should be included in your workspace for section 1 (A2part3.m),

Ts - Sampling period
t - Time domain vector
MUXSIG - Frequency domain representation of mux
f - Frequency vector
fshift - Frequency shifts
Mag - Magnitude
Phase - Phase
xdm - All de-shifted signals in the time domain
XDM - Frequency domain representation of xdm
freqResponse - Frequency response of systems
impresp - Impulse response of chosen system
MSG - Frequency domain representation of Filtered signals
impulses_recov - Recovered impulse responses resampled to 16kHz with DC offset removed
texts_recov - Recovered text streams truncated and with DC offset removed
FS_recov - Sampling rate of the recovered impulse responses at 16kHz
t_recov - New time vector for recovered and resampled signals
f_recov - New frequency vector for recovered and resampled signals

Reflection (CR 3d)

Include a short discussion (maximum 300 words long) that covers any lessons learned and things that you would have done differently. Marks for this are included as part of the criteria available at the end of this brief.

Academic Integrity Declaration (Mandatory)

The provided Academic Integrity Declaration and contribution online form must be completed and submitted along with the assignment. Each student from the group will need to complete their own form. Marks may be moderated depending on contributions. Assignments with incomplete or missing declarations will not be marked. *Familiarise yourself with the university's policy regarding **plagiarism** and **collusion**. See the file "Academic Honesty Slides.pdf" posted with this assignment for some useful details.*

Report and Code Presentation

This assignment includes elements of writing and coding. This is a group assessment item and you are expected to generate and submit:

- One assignment report, "The Report",
- One set of MATLAB code, "The Code" (including at least A2part2.m and A2part3.m), and
- Two data files (A2P2Data.mat and A2P3Data.mat).

The Criteria Reference Assessment (CRA) sheet has the outlines of the marking standards of this assignment.

The teaching team has put together some pointers for you to consider:

The Report (CR3)

An outstanding report demonstrates clear knowledge and understanding of the subject through a combination of visual, mathematical and coding elements. Correct information that is not articulated clearly will attract deductions. **Remember that you are writing to inform.**

- Present the report so it can be understood without reference to the assignment brief.
- Figures or code referenced should be no more than 1 page turn away.
- Avoid the use of "see appendix" and "refer to .m file".
- Full working is required in mathematics-based sections.
- Ensure legibility in any handwritten working.
- Include a title page that states the unit name, unit code, and your names and student ID numbers.
- **Do not include a table of contents, list of figures, nor a list of tables.**

The MATLAB Code (CR 2)

Working MATLAB code is expected to be submitted, alongside your report to Blackboard. The code needs to be executable (in *.m) and **without** run-time errors. No error correction will be made to make your code "run."

Code should be fully commented to describe intent. Quality comments encapsulate your understanding of the topic.

You may use the code provided in the weekly tutorials to check your solutions. However, you are expected to generate your own code for your assignment. Submitting supplied **.p** code as your own work constitutes academic misconduct and will not be awarded any marks.

Code for this assignment will be marked with the assistance of an automated marking system. Ensure that you follow given instructions carefully, including naming conventions. Your code submitted will also be checked for academic misconduct.

Interview

Group interviews will take place (at the discretion of the teaching team) to ensure demonstrated understanding and skills required for this assignment, by the group, and the individual members. You may be selected and contacted to attend an interview if the teaching team requires clarification about how you arrived at your solutions. Interviews will be a casual discussion. These interviews are compulsory and grades are withheld until they are completed. Marks may be deducted for poor demonstration of understanding of content or assignment knowledge. Consult the CRA sheet for the guidelines of what is expected.

Submission Protocol

Assignments are to be submitted in soft-copy through QUT Blackboard in three parts

- **Each group must submit their assignment group plan by 11:59pm on the due date at the top of the brief.**
- A completed academic integrity and group contribution online form. This form is to be completed individually by every student.
- The report. Only **ONE** group member is required to submit the report to the Turnitin link. Coordinate within your group who this will be.
- Your data and code files. Include everything here that your code needs to run. You may submit as either a single zip file, or attach your required files individually.

Some further points:

- Submission deadline is by 11:59pm on due date at the top of the brief.
- This will be a hard deadline, and late submission will not be accepted. As per QUT policy, late assignments receive 0 marks, unless you have applied for and received approval for extension, as per the university policy.
- You do not need to assign your submission with a special name.

- You will need to be registered to a group before you can submit your assignment.
- You may submit as many times as you like before the deadline. New submissions overwrite old submissions. Therefore, only the latest submission will be marked.
- All documents can be reviewed after submission, and thus it is your responsibility to verify the uploaded documents.
- Be aware that the electronic time stamp is placed only after all files have been uploaded successfully.

Don't risk the late penalty and submit early.

Criteria and Standards for Grading

Criteria	Standards						
	CR1. Conceptual understanding 45% Weighting						
	7+	**7**	**6**	**5**	**4**	**3**	**2/1**
CR1a – Theory: Analyse and mathematically **represent** signals, **selecting** Fourier series (**FS**), Fourier transform (**FT**), and/or Laplace transform (**LT**) methods.	Demonstrates exceptional understanding of the theoretical concepts of the FS, FT, and LT and their properties, through the use of graphical and mathematical methods. That are clear, accurate and logical (with no errors). • Derives and expresses correct mathematical relationships between signals' time and frequency representations, using Fourier and Laplace analysis.	Demonstrates accurate understanding of the theoretical concepts of FS, FT, and LT and their properties, through the use of graphical and mathematical methods that are clear, accurate and logical (with one minor error): • Derives and expresses correct mathematical relationships between signals' time and frequency representations, using Fourier and Laplace analysis.	Demonstrates mostly correct understanding of the theoretical concepts of FS, FT, and LT and their properties, using graphical and mathematical methods used are clear and logical (No critical errors, but with a couple of minor errors or one theoretical error): • Derives and expresses correct mathematical relationships between signals' time and frequency representations, using Fourier and Laplace analysis.	Demonstrates sound understanding of the theoretical concepts of FS, FT, and LT and their properties using clear and accurate graphical and mathematical methods (No critical errors, but with a couple of minor errors or one theoretical error): • Derives and expresses correct mathematical relationships between signals' time and frequency representations, using Fourier and Laplace analysis.	Demonstrates adequate understanding of the theoretical concepts of FS, FT, and LT and their properties, using logical graphical and mathematical methods (with many minor representation errors, a few theoretical errors or two critical theoretical errors): • Derives and expresses mostly correct mathematical relationships between signals' time and frequency representations, using Fourier and Laplace analysis.	Shows minimal or incoherent understanding of theoretical concepts of FS, FT, and LT and their properties (Explanations or calculations have major errors. Graphical representations missing / major technical errors). • Attempts to derive and express the relationships between signals' time, frequency and Laplace representations.	No evidence of understanding of theoretical concepts. Few or no calculations (Incorrect interpretation, signal representations or derivations.) • No attempt to identify Fourier and Laplace characteristics.
CR1b – Maths Analysis: Investigate the relationships between time and frequency domain signal representations.	• Utilises the mathematical relationship between real and imaginary, FT magnitude and phase, with no errors. • Correctly evaluates FS, FT, LT and appropriately uses their characteristics, and identifies effects of shifted spectra, with no errors. • Performs all mathematical simplifications necessary.	• Utilises the mathematical relationship between real and imaginary, with a couple minor errors. • Correctly Evaluates FS, FT, LT and appropriately uses their characteristics, and identifies effects of shifted spectra, with a couple minor errors. • Performs almost all mathematical simplifications necessary.	• Utilises the mathematical relationship between real and imaginary, with a few minor errors or one misapplication of the relationships. • Evaluates FS, FT, LT and appropriately uses their characteristics, and identifies effects of shifted spectra, with a few minor errors or one misapplication of the relationships.. • Performs most mathematical simplifications.	• Utilises the mathematical relationship between real and imaginary, with a several minor errors or a couple misapplications of mathematical relationships. • Evaluates FS, FT, LT and appropriately uses their characteristics, and identifies effects of shifted spectra, with a several minor errors or a couple misapplications of mathematical relationships. • Performs many mathematical simplifications.	• Utilises the mathematical relationship between real and imaginary, with many minor errors or a few misapplications of mathematical relationships. • Comments on FS, FT, LT and uses their characteristics, and identifies effects of shifted spectra. • Performs a few mathematical simplifications.	• Attempts to Utilise the mathematical relationship between real and imaginary, with several major errors. • Fails to evaluate or comment on signals' spectra, • Performs no mathematical simplifications.	• Does not correctly utilise or apply mathematical relationships to given signals. • No comments on signals' representation or spectra.

CR1c - Interpret the input/output relationships of Linear Time Invariant (**LTI**) systems.	• Accurately and correctly represents and analyses LTI systems with no errors. • Performs convolution/multiplication operations in the time and frequency domains with no errors.	• Represents and analyses LTI systems with a couple minor errors • Performs convolution/multiplication operations in the time and frequency domains with a couple minor errors.	• Represents and analyses LTI systems, with a few minor errors or one misapplication of the relationships • Performs convolution/multiplication operations in the time and frequency domains with a few minor errors or one misapplication of the relationships.	• Represents and analyses LTI systems with a several minor errors or a couple misapplications of mathematical relationships. • Performs convolution/multiplication operations in the time and frequency domains with a several minor errors or a couple misapplications of mathematical relationships.	• Represents and analyses LTI systems with many minor errors or a few. misapplications of mathematical relationships. • Performs convolution/multiplication operations in the time and frequency domains with many minor errors or a few. misapplications of mathematical relationships.	• Attempts to represent and analyse LTI systems with several misapplications of mathematical relationships. • Attempts to perform convolution/multiplication operations in several errors exist.	• No evidence of ability to represents or analyse LTI systems or perform convolution/multiplication operations.
CR1d - Select and justify the use of appropriate filters.	• Select appropriate filters based on their transfer functions and the relationships between the impulse and frequency responses, with evidence of further research with appropriate references. • Thoroughly justifies both quantitatively and qualitatively, the selection of parameters, methods, and techniques.	• Select appropriate filters based on their transfer functions and the relationships between the impulse and frequency responses. • Thoroughly justifies the selection of parameters, methods, and techniques with one minor error.	• Select appropriate filters based on their transfer functions and the relationships between the impulse and frequency responses. • Justifies the selection of parameters, methods, and techniques with several minor errors.	• Select appropriate filters based on their transfer functions and the relationships between the impulse and frequency responses. • Justifies the selection of parameters, methods, and techniques with one major error, several minor errors, or has omitted one necessary justification.	• Select appropriate filters based on their transfer functions and the relationships between the impulse and frequency responses. • Justifies the selection of parameters, methods, and techniques with a few errors or has omitted a couple necessary justifications.	• Attempts to select filters. • Mostly fails to correctly justify the selection of parameters, methods, and techniques.	• No evidence of selecting required filters. • Fails to correctly justify any selection of parameters, methods, and techniques.
	CR2. Application using MATLAB Coding 35% Weighting						
	7+	**7**	**6**	**5**	**4**	**3**	**2/1**
CR2a - Design effective code to represent signals and spectra using FS, FT and LT and **generate** time and frequency vectors.	• Well-commented MATLAB code with all appropriate simplifications and optimisations performed. • MATLAB functions have been developed to optimise code for repeated processes.	• Well-commented MATLAB code with most of the appropriate simplifications. • Most optimisations are performed.	• MATLAB code is commented throughout. • Many simplifications and optimisations are performed.	• MATLAB ode is commented. • Some simplifications and/or optimisations are performed.	• MATLAB code has some structured approach but is difficult to follow. • Only occasional simplifications performed.	• MATLAB code is unstructured. • Code is difficult to follow and exhibits major flaws.	• Little attempt is made to write MATLAB code.
CR2b – Evaluate and **interpret** signal representations and spectra using Fourier and Laplace analysis and supplied functions.	• Correctly calculates all required variables with no errors.	• Correctly calculates all required variables with only a couple minor errors.	• Correctly calculates all required variables with a few minor errors.	• Calculates all required variables with several minor errors or one major error.	• Calculates required variables with many minor errors or a couple of major errors. There must be no run time errors in the code.	• Attempts to calculate required variables with more than a couple major errors or run time error.	• No attempts to calculate required variables.

	CR3. Effective written communication 15% Weighting						
	7+	7	6	5	4	3	2/1
Assignment solution is presented in a Report format considering: **CR3a - Structure, CR3b –Integration, CR3c – Figures, graphs and tables CR3d – Reflections and Contextualisation.**	• Professional report format, that is easy to read and core technical ideas are clearly and accurately conveyed. All processes and solution steps are clearly outlined. • Code is seamlessly integrated into the report and are used effectively to convey ideas in the report. Code snippets are selectively chosen to succinctly demonstrate all necessary key technical ideas. • Figures are effectively used throughout the report to demonstrate results and facilitate explanations. Figures are labelled and plots are appropriately scaled and visible. • Solution is contextualised to the MARS242 mission, and includes **reflections** conveying accurate technical knowledge and insights.	• Professional report format, that is easy to read and core technical ideas are clearly and accurately conveyed. Almost all appropriate processes and solution steps are clearly outlined. • Code is integrated into the report and are used to convey the ideas in the report. Selected code snippets demonstrate all key technical ideas. • Figures are effectively used throughout the report to demonstrate results and facilitate explanations. Figures are labelled and plots are appropriately scaled and visible. • Solution is contextualised to the MARS242 mission, and includes **reflections** conveying accurate technical knowledge.	• Professional report format, that is easy to read and core technical ideas are conveyed. Most of the appropriate processes and solution steps are clearly outlined. • Code is integrated into the report and are used to convey ideas in the report. Selected code snippets demonstrate most key technical ideas. • Figures are used throughout the report to demonstrate results and facilitate explanations. Figures are labelled and plots are appropriately scaled and visible. • Solution is contextualised and includes **reflections**. conveying accurate technical knowledge.	• Report is mostly easy to read and most core technical ideas are conveyed. Some key explanations of processes and solution steps are missing. • Code is integrated into the report and are used to convey ideas in the report. Code snippets are used. • Figures are used to demonstrate results and facilitate explanations. Most figures are labelled and plots are appropriately scaled and visible. • Solution is contextualised and includes **reflections**.	• Report is missing some key components of a report. Report is difficult to read in some parts and some core technical ideas are conveyed. Many key explanations of processes and solution steps are missing. • Code is not integrated into the report or is not used to connect with the ideas discussed in the report. • Figures are included but do not demonstrate results or facilitate explanations. Some important figures are missing. • Solution contains some contextualisation and **reflections**.	• Report is missing multiple key components of a report. Report is difficult to read. A significant number of key explanations of processes and solution steps are missing. • Code is not integrated and is mostly a code and figure dump. • Multiple key figures are missing and little effort has been made to link the figures with the report. • Attempts to contextualise the solution but with no **reflections**.	• Report has little to no structure. Large portions of the report are missing/not attempted. • Code is incorrect or missing from the report. • Figures are almost entirely missing from the report. • No contextualisation or **reflections**.

	Assignment Plan 5% Weighting	
	Full **5%** is given if: Assignment **plan** is completed, signed by all students, and submitted by the deadline.	Plan not completed before the deadline.

For moderation of overall marks: At the discretion of the teaching team, your group may be selected to attend an interview if the teaching team require clarification about how the group arrived at their solutions, or how individuals contributed to the overall solution. If selected, you will be notified and given details of the location and time of the interview.

Oral interview - Group Post-submission theoretical and applied skills.	Demonstrate excellent theoretical and applied knowledge retention after assignment submission. All members of the group must be able to correctly explain theory and justify methods used in their submission including MATLAB code.	Demonstrate adequate knowledge retention. If a student fails to show adequate understanding of their assignment and its components, including MATLAB code, his/her overall mark for this assignment may be moderated down by a maximum of 20%.	Demonstrate inadequate knowledge retention. If a student does not attend the interview, or fails to demonstrate fundamental knowledge of their submission and its components, including MATLAB code, his/her overall mark may be moderated By up to 100%.