

Engineering Tripos Part IIA Project, GF3: Audio Modem, 2020-21

Leader

[Dr J Sayir](#) [1]

Timing and Structure

Online supervisions Thursdays 11-1pm and Mondays 9-11am, Project Standardisation meeting (one delegate per group) every day 10-11 am

Prerequisites

IB Paper 6 assumed (Discrete Fourier Transform!), 3F1, 3F7 and/or 3F4 useful but not assumed

Aims

The aims of the course are to:

- design a software modem that transmits files from one computer to another over the audio system (loudspeaker, microphone)
- gain a practical understanding of modulation and coding by implementing them on real-world channels and data
- gain a comprehensive view of all elements involved in communication systems, including several not taught in our lectures, e.g., synchronisation and channel estimation
- deepen your understanding of the discrete Fourier transform (DFT) through the implementation of a Baseband-OFDM / DMT modulation system
- learn to design software in Python and/or other programming languages

Content

Communications engineering can be a pain to convey to the general public because the border between abstract concepts and concrete products is hard to bridge: you design methods using math, sometimes test them in simulations, but they cannot become concrete without specialised hardware. Once the product is made, it's hard to point your finger and tell your friends "Here: I made this part of the product!". This project is different: you are to design a full modem for data transmission over audio, where files will be transmitted from one computer over its loudspeaker and received by another through its microphone. In a way, we are going back to the good old days of telephone modems where users could hear a painful string of beeps and squeaks when they transferred files, except that we will use state-of-the-art modulation and coding techniques that did not exist back then, and achieve data rates and reliability no one has ever achieved before.

The end product should be a working piece of software with the suggested name "Cambridge Audio Transfer" (cat logos and cat "meeows" to announce the start of transmission would be a nice finishing touch!) The software should use a variant of Orthogonal Frequency Division Multiplexing (OFDM) as its modulation (taught in 3F4), and Low-Density Parity-Check (LDPC) codes as part of its coding (taught in 3F7). If you have not taken these modules, you will be able to learn the techniques during the project as they are easy to grasp, but you won't get the more in-depth understanding provided in the lectures. In addition, you will have the opportunity to learn about synchronisation, channel estimation, and other modem components during the project.

Note that this is a real "project" in the sense that it is an open-ended undertaking whose outcomes differ from year to year and from group to group depending on who's participating, what technical decisions each group takes and on what framework the groups agree during standardisation meetings. If you are

looking for a scripted exercise where the whole project is determined in advance and everyone does exactly the same, then this project may not be for you. The project will appeal to engineers who want the freedom to be creative and to find original solutions to problems. Those who approached the project with this mindset in past years (about 90% of participants) have loved the project.

The subject matter of this project is communications and signal processing. The project is a natural choice for those doing information-based engineering areas but we've had participants from other areas in the past who did very well and enjoyed the project. Essentially, it's an exercise in understanding and using the Discrete Fourier Transform in all its details, which is relevant for mechanical, civil, aeronautical and electrical engineers just as much as it is for information engineers.

FORMAT

You will work in groups of 3. Initially, you will be guided as a group through multiple levels and challenges. Later it will be part of your assignment to find an appropriate task distribution that mixes parallel integration (you work on this part, I work on that part of the system) with vertical integration (how to dimension the overall system). The project is run partly online (supervision, group collaboration on slack, some demos including final "plugfest" online) and in-person meetings can take place in accordance with the safety requirements in place in Easter term (some standardisation meetings in person, and an extra final demon in person.)

Technical requirements: you must have access to at least 1 computer and 1 smartphone. If you have a 2nd computer, that's a bonus but not a requirement. Also, if you are able to collaborate in person (group members from the same household) that's also a bonus.

Grouping: the groups will be arranged just before the start of the project taking into account your preferences and a number of criteria (household groups will be prioritised, each group to have at least one proficient programmer and if possible someone who has taken 3F7/3F4).

Week 1: *Quick & Dirty implementation phase*

Complete the missing parts in a framework software. Students will be guided step by step through the process of implementing an OFDM/DMT modulator and demodulator, to work with provided basic channel estimation, equalisation, synchronisation to produce a first working modem. Aim to achieve successful transmission of a file over a simulated channel.

Week 2: *Optimisation phase*

Perform channel measurements and develop synchronisation techniques. This is the hardest part of the project and your conclusions will feed directly into the standardisation

capacity computations. Learn about unequal loading of constellations in OFDM, waterfilling optimisation, LDPC code design using density evolution and/or EXIT charts, irregular Repeat-Accumulate IRA codes, and/or Turbo codes.

Week 3: *Advanced implementation*

Refine the DMT implementation, adapt channel equalisation, implement techniques to optimise the loading and power profiles. Design an LDPC weight polynomial and generate corresponding encoder and parity-check matrices. Implement an IRA encoder / decoder. Implement a Turbo encoder / decoder using convolutional codes. Start planning report.

Week 4: *Final refinements*

Improve synchronisation, implement real-time channel estimation using pilot symbols, test transmission under various conditions (simulated channel, anechoic quiet outdoor scenario, indoor with echo, indoor with echo and interference.) Measure capacity, data rate and error rates achieved. Test transmission in restricted frequency ranges to emulate the regulatory masks that dictate the design of wireless systems. Write report.

Coursework

Coursework	Due Date	Marks
Standardisation contributions	throughout project	20 (group)
Interim report (2 pages)	Wednesday, 26 May, 4 pm	20 (individual)
Draft Final Report	Sunday, 6 June, end of day	0
Final demo and presentation	Monday, 7 June, 9:15 am	20 (group)
Final report	Thursday, 10 June, 4 pm	30 (individual)

Examination Guidelines

Please refer to [Form & conduct of the examinations](#) [2].

Last modified: 03/12/2020 08:53

Source URL (modified on 03-12-20): <https://teaching.eng.cam.ac.uk/content/engineering-tripos-part-ii-a-project-gf3-audio-modem-2020-21>

Links

[1] <mailto:js851@cam.ac.uk>

[2] <https://teaching.eng.cam.ac.uk/content/form-conduct-examinations>