

SYLLABUS

BIOS 214: Open-Source MRI Sequence Development Simulation to Scanner

Course information

Dates	November 9–December 4, 2026 · Mondays, Wednesdays, and Fridays
Times	📅 Lecture: 9:30–10:20 a.m. · 🖥️ Laboratory: 10:30–11:50 a.m.
Locations	📍 Classroom: CCSR4107 , except days 1 and 3: M218/218A . 📅 Friday laboratory: Lucas Center — Cima.X (3T3).
Instructors	Carlos Castillo-Passi, Ph.D. , Postdoctoral Scholar in Radiology, Stanford University. Daniel Ennis, Ph.D. , Professor of Radiology and Bioengineering (by courtesy), Stanford University.
TA	Charles McGrath, Ph.D. , Postdoctoral Scholar in Radiology, Stanford University.

Course links: [Course website](#) (schedule/materials) · [Canvas](#) (announcements/submissions)

Course overview

This interdisciplinary mini-course introduces an end-to-end workflow for MRI sequence development using open-source tools. Students will design pulse sequences in [Pulseseq](#), simulate and validate them with [KomaMRI](#), and reconstruct simulated or acquired data using [BART](#) and [MRIReco](#). Hands-on laboratory sessions connect simulation to scanner experiments through [Pulseseq](#) interpreters and [OpenRecon](#), with an emphasis on reproducible workflows and scientific interpretation.

There will be an emphasis on research rigor, reproducibility and transparency, by using open-source tools and the creation of reproducible pipelines. Intended for students with basic programming experience and an interest in MRI and computational methods.

Learning goals

By the end of this mini-course, students will be able to:

1. Design basic MRI pulse sequences using [Pulseseq](#).
2. Simulate MRI experiments using Bloch simulations in [KomaMRI](#).
3. Reconstruct MRI data using [BartIO](#) and [MRIReco](#).
4. Analyze the impact of sequence parameters on image contrast and artifacts.
5. Implement and iterate on sequence designs using a rapid prototyping workflow.
6. Evaluate sequence performance through simulation and scanner experiments.

Before the first class

Complete these steps:

👤 yourself; 🤖 coding agent

1. 👤 Install and sign in to a **desktop coding agent** (not the CLI version), such as [Codex](#) or [Claude Code](#).
2. 🤖 Ask your coding agent to install [Julia 1.12](#) and add [Pluto](#) to Julia's global environment.
3. 🤖 Ask your coding agent to clone the [activities repository](#).
4. 👤 **Open the cloned folder as a project in your coding agent** to access AGENTS.md and the course skill.
5. 🤖 Ask your coding agent to open **Activity 0**, then 👤 complete the activity to familiarize yourself with the environment and confirm:
 - 👤 Correct environment setup, including Julia 1.12 (check VERSION).
 - 👤 Pluto's reactivity when inputs change.
 - 👤 Use of the course-provided skill file by your coding agent.

Report any setup problems before the first class. This readiness check is ungraded.

Assessment and grading

Grading uses a point system, with points earned through attendance and completed laboratory activities.

Activity	Points each	Number	Total
Lecture attendance (33.3%)	2	9	18
Completed regular laboratory (33.3%)	3	6	18
Completed scanner laboratory (33.3%)	6	3	18
Total			54

Required to pass: 45 points.

- **Lecture attendance:** Attend and record attendance using the Canvas check-in form.
- **Coding laboratory:** Use the skill; submit a reproducible notebook with expected results and prompt history.
- **Scanner laboratory:** Prepare and scan a Pulseq sequence; compare measurements with simulations.

Submission deadline: Submit laboratory work through Canvas by **midnight at the end of the day**.

Additional points: Giving back to the projects we use is a central part of open source. Contributions earn additional points that count toward the passing requirement:

- **1 point** for an issue, bug, or enhancement report with a minimal working example (MWE).
- **3 points** for a pull request approved through the project's review process.

Course schedule

Each class day includes a lecture followed by a laboratory session. Monday and Wednesday sessions focus on simulation and coding; Friday sessions bring the week's work to the scanner.

Week	Monday	Wednesday	Friday
Week 1 RF excitation	Day 1 · Nov 9 📖 Lecture: Open source, AI agents, and MRI simulations 💻 Laboratory: Run and interpret a first Bloch simulation	Day 2 · Nov 11 📖 Lecture: RF excitation and selective pulses 💻 Laboratory: Simulate the 2D RF excitation sequence	Day 3 · Nov 13 📖 Lecture: RF pulse validation 🏠 Scanner laboratory: Test your chosen 2D RF pattern; compare with simulations
Week 2 Spatial encoding	Day 4 · Nov 16 📖 Lecture: Gradients and Cartesian k-space 💻 Laboratory: Simulate gradient moments and EPI artifacts	Day 5 · Nov 18 📖 Lecture: Radial and spiral trajectories 💻 Laboratory: Compare undersampling, delays, and off-resonance	Day 6 · Nov 20 📖 Lecture: Trajectories and hardware limits 🏠 Scanner laboratory: Acquire and reconstruct radial data
Week 3	📅 November 23-27 · Thanksgiving break		
Week 4 Sequences and reconstruction	Day 7 · Nov 30 📖 Lecture: Gradient echo, spin echo, and steady states 💻 Laboratory: Simulate spoiling, banding, and relaxation	Day 8 · Dec 2 📖 Lecture: Coil sensitivities, parallel imaging, and compressed sensing 💻 Laboratory: Reconstruct data using BART and MRIReco	Day 9 · Dec 4 📖 Lecture: Pulseseq and OpenRecon 🏠 Scanner laboratory: Evaluate an acquisition and reconstruction pipeline

Classroom: ☺ [CCSR4107](#), except days 1 and 3: ☺ [M218/218A](#).

AI policy

AI use is encouraged in this course. You may use AI to generate all code, debug implementations, produce visualizations, and explore concepts. The goal is to reduce programming barriers so you can focus on MRI physics, experimental design, and interpretation.

AI can help you find information and write code, but **it does not replace understanding. We are here to learn!** You must understand what your submitted code does and how it relates to the physics being studied. You should be able to explain your assumptions, justify your design choices, predict outcomes, and validate results.

You are responsible for any errors in your submitted work, including those introduced by AI tools. Assessment will emphasize scientific understanding and your ability to explain and evaluate your work, and may include individual questions or discussions without AI.

Honor Code

This course follows [Stanford's Honor Code](#) and promotes academic honesty and integrity.