

Getting started

This page: structure of this subsystem and where to find what.

Make sure you understand the embedded structure: [Embedded Infastructure](#)

1) Main

Main.c can be found in `src/arm_board`

This is the code that will be ran when building and uploading using platformio. Main should contain some multithreading for running separate tasks, otherwise it should use the libraries (common and arm board specific) that are created.

2) Libraries

The libraries for the arm board can be found in `components/arm_board`

The arm board uses 3 libraries:

- firmware
- movement
- simulink

Firmware contains the generated CubeMX code. **You do not need to touch this after generating.**

NOTE: do make sure that after generating your code in CubeMX, you run the post generation script. You *can* set a post generation script in CubeMX itself. However, if you use **Windows** you may need to run the bash script manually. This script can be found in `scripts/post_code_generation.bash`.

Simulink contains code generated by control subteam. You also do not need to touch this, since it is not code that is ran on embedded side. It is a helpful **reference** for the output data that control will be giving your system.

Specifically, in `control.h`, struct `ExtY` gives the external outputs. These will be transferred across the robot using protobufs. So for us, this struct contains the inputs for the motors on the robotic arm.

Movement is currently the only "real" library that is written by hand. It contains the source code for controlling stepper motors.

3) Protobuffers

Information on the arm board protobuffers can be found here: [Arm Board Protobuffers](#)

Revision #17

Created 2026-04-14 08:19:56 UTC by Lisa te Braak

Updated 2026-04-22 10:36:17 UTC by Lisa te Braak