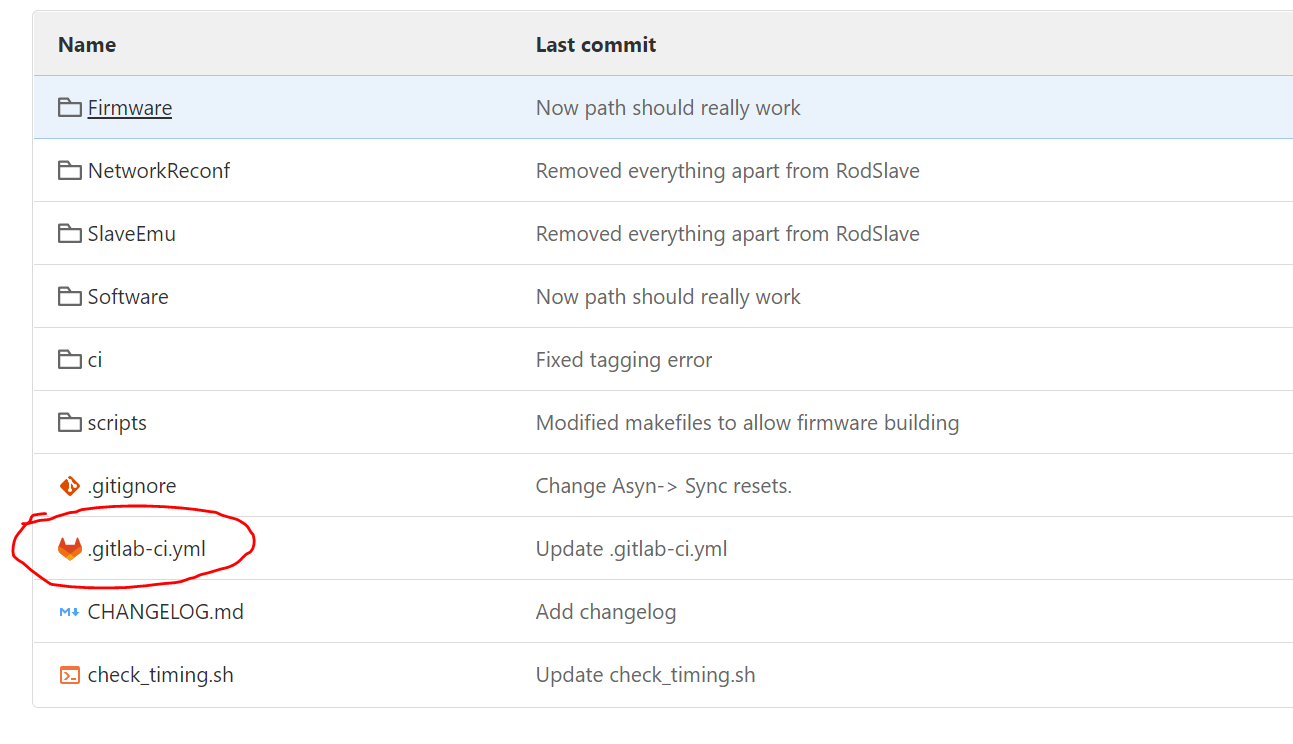
**Tutorial 7**

**Gitlab CI/CD Pipeline for Firmware generation**

*November 2022*

The CI/CD feature in GitLab is used to generate firmware from the VHDL and constraints provided. This document does not cover the details of that but only gives and overview of how it is being used in the RodSlave repo for ATLAS Pixel.

.gitlab-ci.yml file describes the stages and which scripts to execute in each stage.



These are scripts for building firmware and bitfile generation.

For more info on the stages and scripts refer to : <https://github.com/uw-acme/acme-lab-documentation/blob/main/lhc/Pixel_IBL/Smart_L1A/related_material/18_Firware_WorkFlow_releases.pdf>

The pipeline of these stages is set to trigger after a merge request however this can be changed to have the option to manually trigger a pipeline without a merge by editing the .yml file.

In this system the basic idea is to build firmware, test it (manual as of now) and then deploy (put bitfiles into cvfms folder)

This setup is useful because it allows us to use the tools in the machines at CERN, so we do not need Xilinx tools in our personal setup. It also builds FW for both north and south FPGAs with their respective ucf files (they are different because of termination issues in the BOC board!!) which is time consuming and prone to errors otherwise.

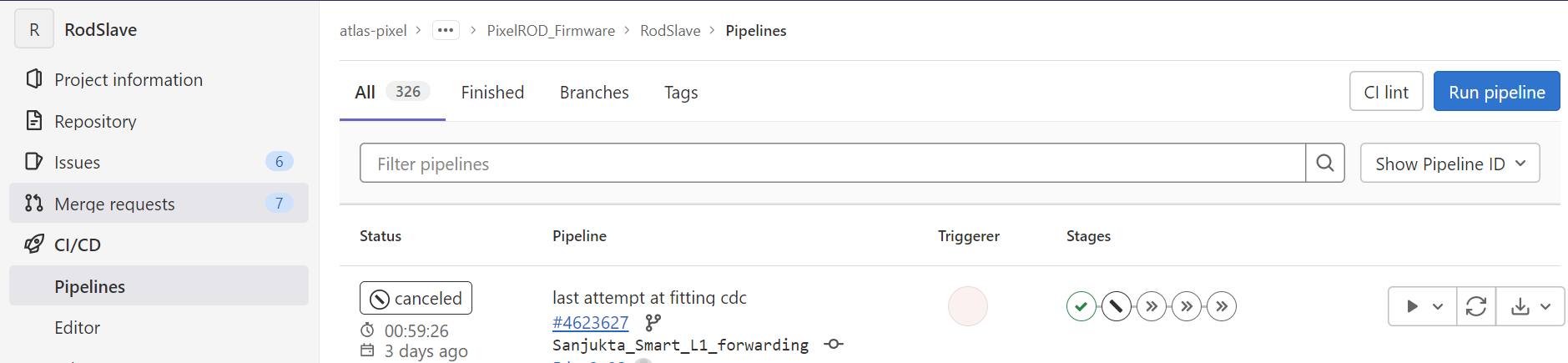
The log can be seen while the job is running, things like timing score and progress are useful to look at.

**How to use:**

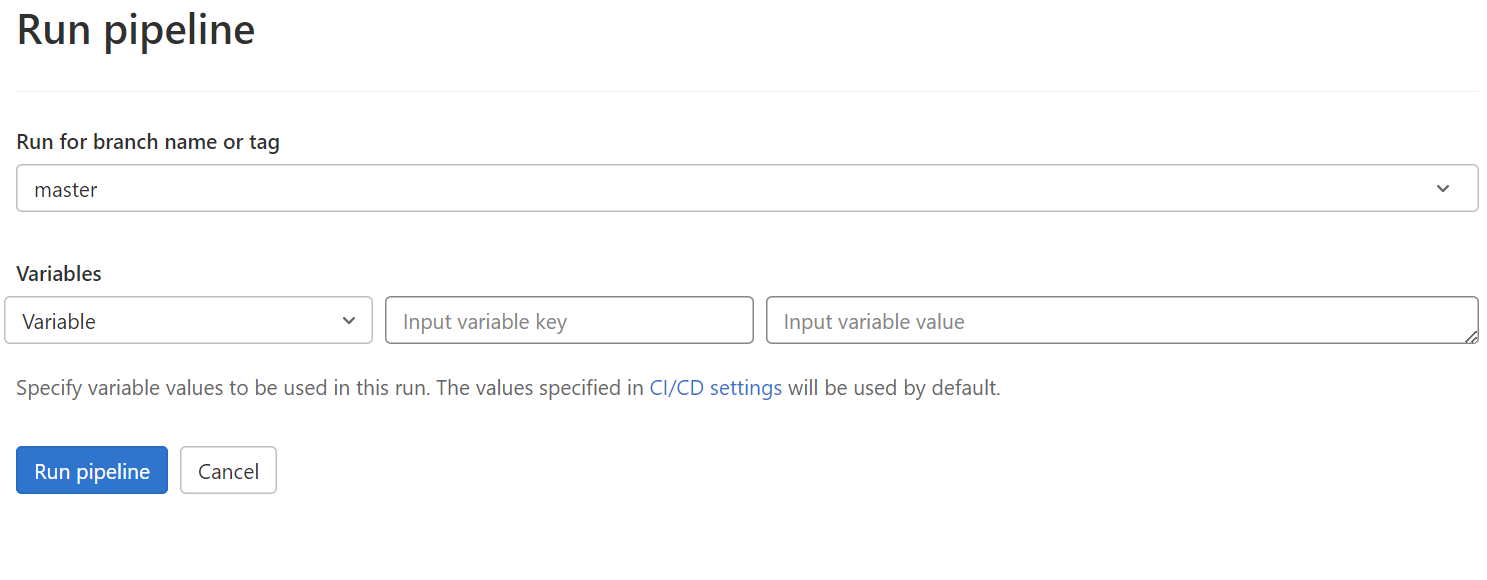
Commit and push changes in your branch

Create a merge request (pipeline should run automatically)

To view pipeline and status go to pipelines under CI/CD and you should see the pipelines and job running



To manually trigger a pipeline, go to run pipeline in the above image. Select your branch and click run pipeline. Variables can be empty here as there are none. You should then be able to see the triggered pipeline as above.



If the build stage is successfully completed (no failures, it fails if there is an unmet timing constraint for example).

This is set up to generate bitfiles for flavours of fw of these combinations- Pixel/IBL, datataking/calibration, and with/without ChipScope (CS) ILA. Chipscope works in the same way as described in the previous tutorial, the cdc file is considered during implementation in these scripts.

The versions with CS are allowed to fail (i.e. the next stage will be triggered even if they fail), but it is good practice to get those to be functional as well so that any future debugging is possible.

If all flavors pass, they are deployed into the cvfms folder accessible in testbed machines.

Even if they all do not pass, the outputs (bitfiles, ace files) are still available to download by clicking into each. These bitfiles can be loaded into slave FPGAs as described in previous tutorials.

