

GitLab Portal

<http://cas.polito.it/gitlab>



**POLITECNICO
DI TORINO**

Groups organization



- 📁 **C** **Course Resources**
Projects and resources for the Computer Architectures course

- 📁 **N** **NXP LandTiger**
LANDTIGER board based on NXP LPC1768 SoC embedding ARM Cortex-M3 – Special Projects

- ✓ 📁 **I** **IFX Aurix**
AURIX board based on INFINEON TRICORE multiprocessor SoC – Instrumented Bike project
 - 📁 **D** **Display**

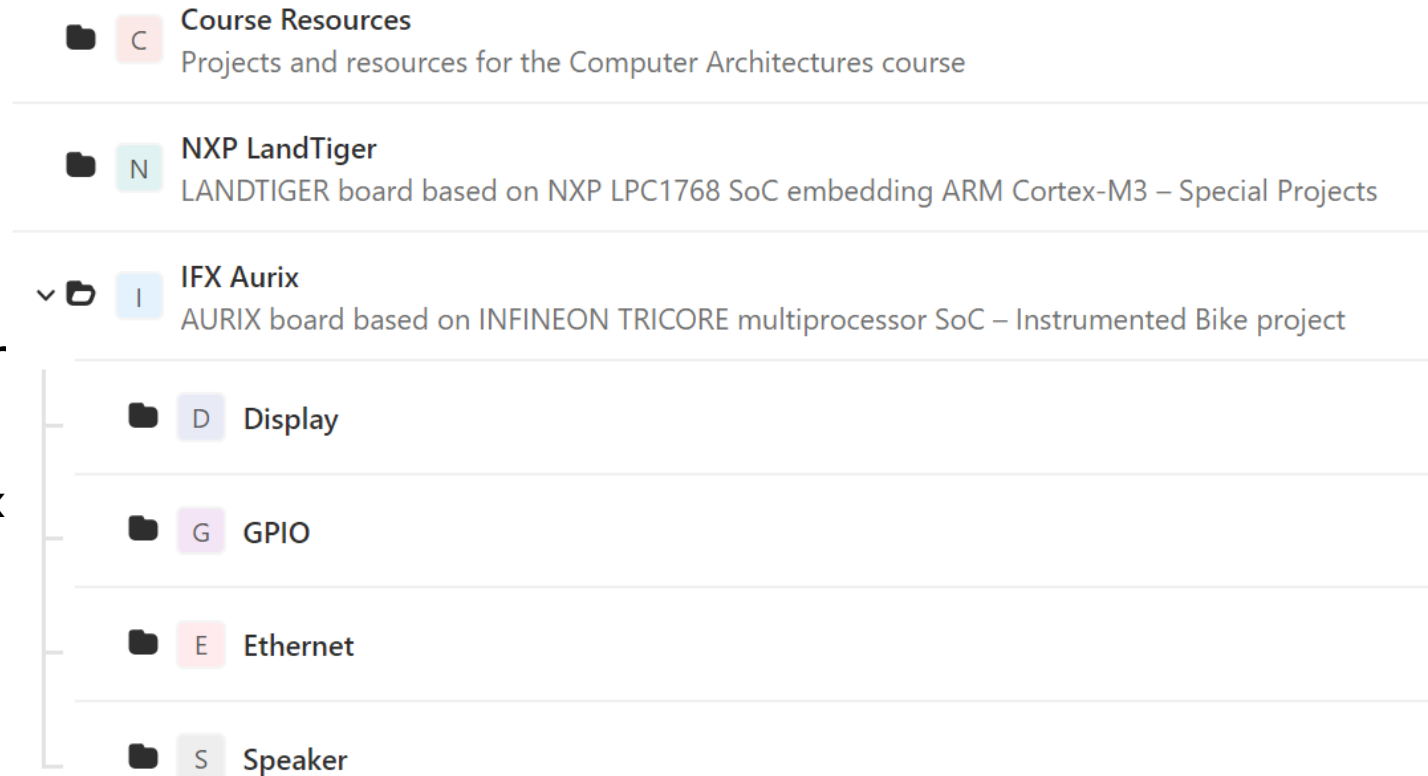
 - 📁 **G** **GPIO**

 - 📁 **E** **Ethernet**

 - 📁 **S** **Speaker**

> IFX Aurix – PERMISSIONS

- Main group with all of us
- **Reporter** is the default user permission for everyone:
 - Visualize all projects in IFX Aurix and subgroups
 - Download/pull the repository
 - Fully manage issues (report bugs...)

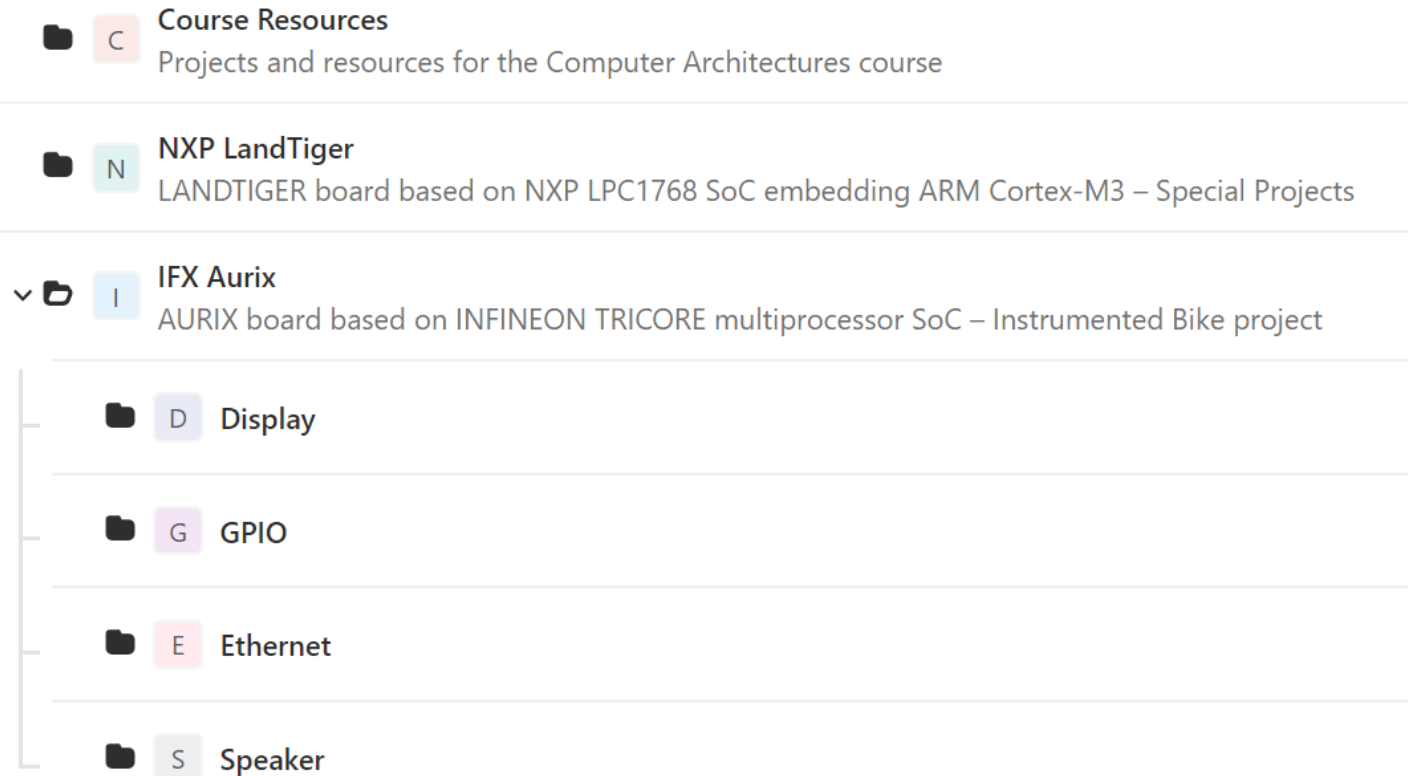


The screenshot shows a GitLab Groups organization structure. It features a tree view with the following groups and their descriptions:

- Course Resources** (C): Projects and resources for the Computer Architectures course
- NXP LandTiger** (N): LANDTIGER board based on NXP LPC1768 SoC embedding ARM Cortex-M3 – Special Projects
- IFX Aurix** (I): AURIX board based on INFINEON TRICORE multiprocessor SoC – Instrumented Bike project
 - Display** (D)
 - GPIO** (G)
 - Ethernet** (E)
 - Speaker** (S)

> Subgroups – PERMISSIONS

- **Developer** permission in the group which you belong to:
 - All Reporter permissions
 - Fully manage non-protected branches of existing projects
 - Manage merge requests on non-protected branches
 - **Cannot create new projects**

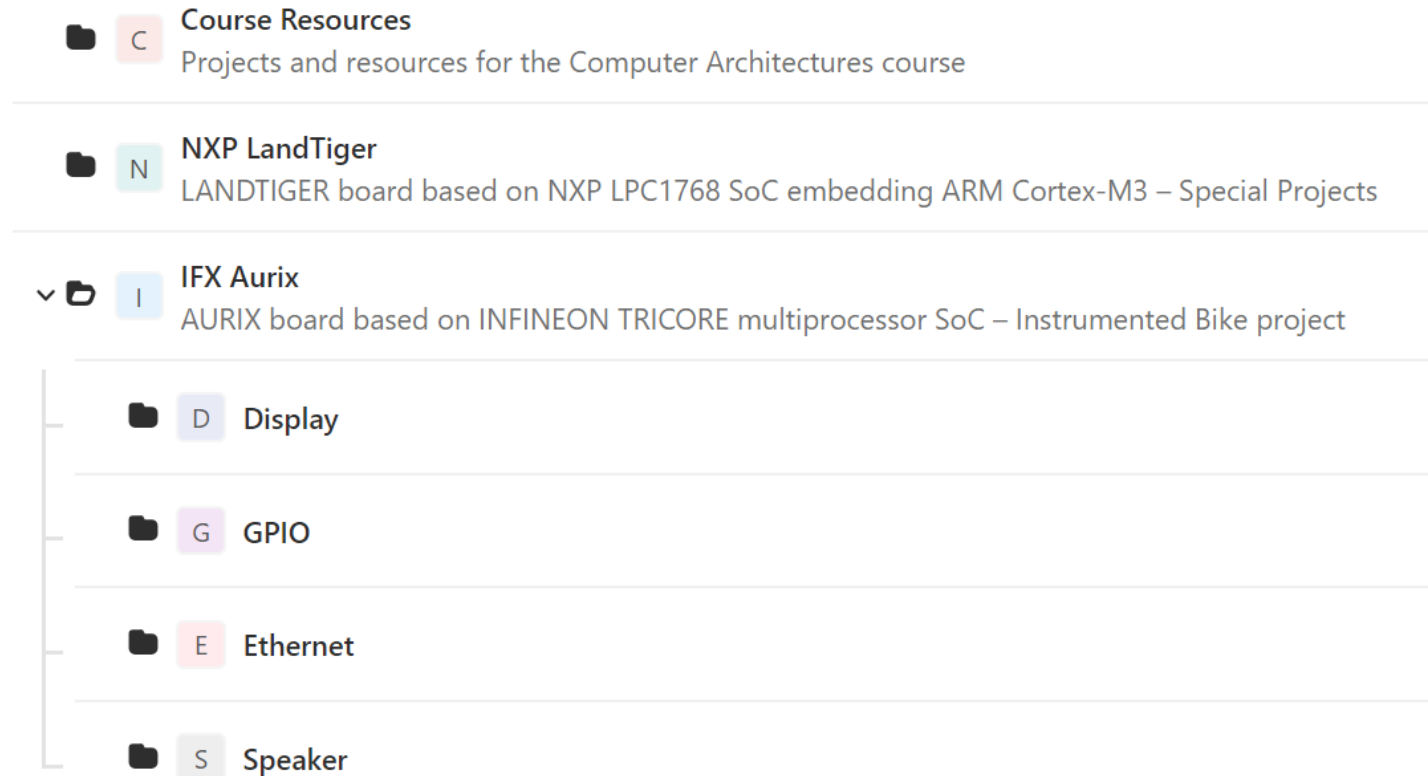


The screenshot displays a GitLab group structure with the following subgroups:

- Course Resources** (C): Projects and resources for the Computer Architectures course
- NXP LandTiger** (N): LANDTIGER board based on NXP LPC1768 SoC embedding ARM Cortex-M3 – Special Projects
- IFX Aurix** (I): AURIX board based on INFINEON TRICORE multiprocessor SoC – Instrumented Bike project
 - Display** (D)
 - GPIO** (G)
 - Ethernet** (E)
 - Speaker** (S)

> IFX Aurix – CONTENTS

- Specific hardware **subgroups**
- **Projects** for the basic working environment + main releases

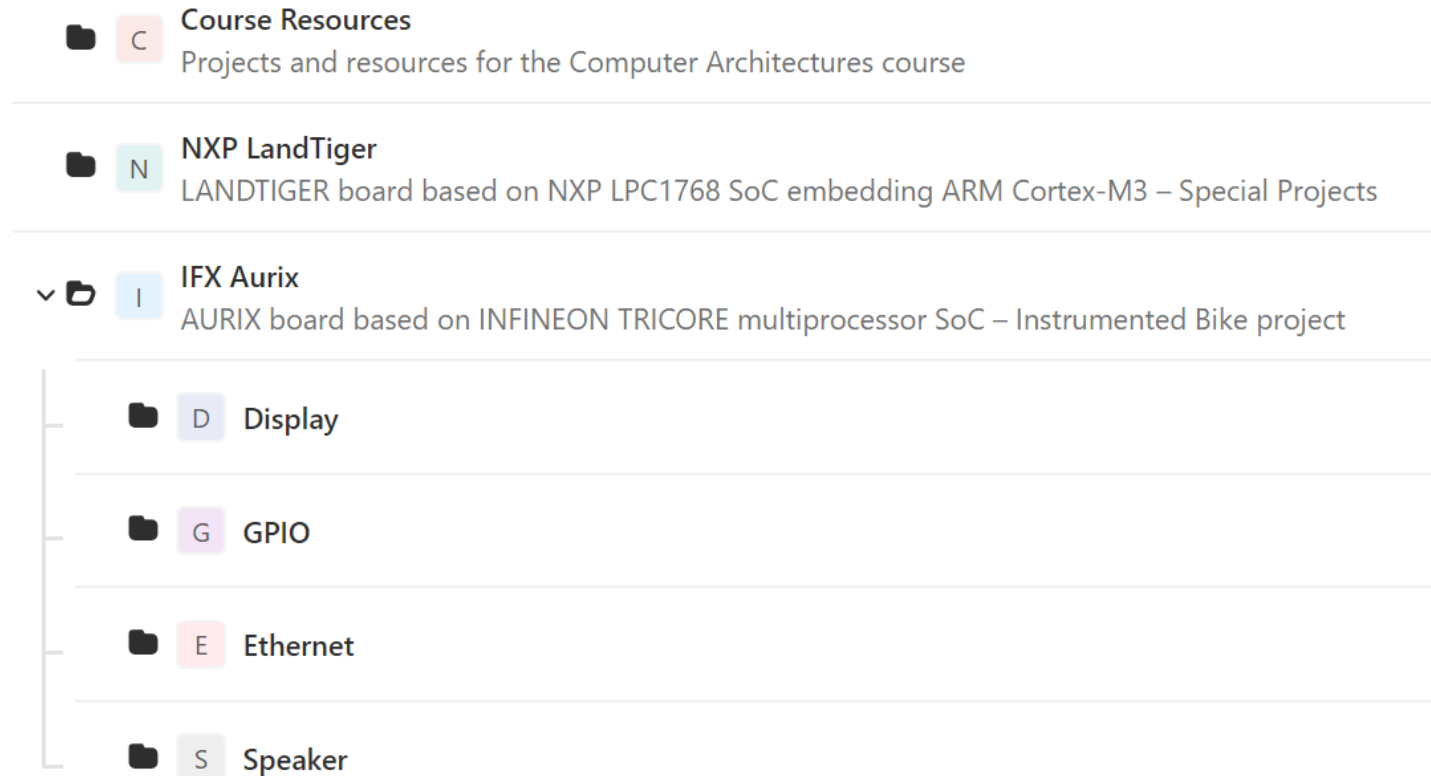


The screenshot displays a GitLab repository structure with the following folders and descriptions:

- C** Course Resources
Projects and resources for the Computer Architectures course
- N** NXP LandTiger
LANDTIGER board based on NXP LPC1768 SoC embedding ARM Cortex-M3 – Special Projects
- I** IFX Aurix
AURIX board based on INFINEON TRICORE multiprocessor SoC – Instrumented Bike project
 - D** Display
 - G** GPIO
 - E** Ethernet
 - S** Speaker

> Subgroups - CONTENTS

- A project with a master protected branch
 - The master branch is a stable version of the ongoing project
 - When a new stable version is available (e.g. updated libraries, new functionalities, ...), do a **pull request** to such master branch (comment, discuss, motivate...)
 - It is possible to have multiple projects in the subgroup: ask us if needed

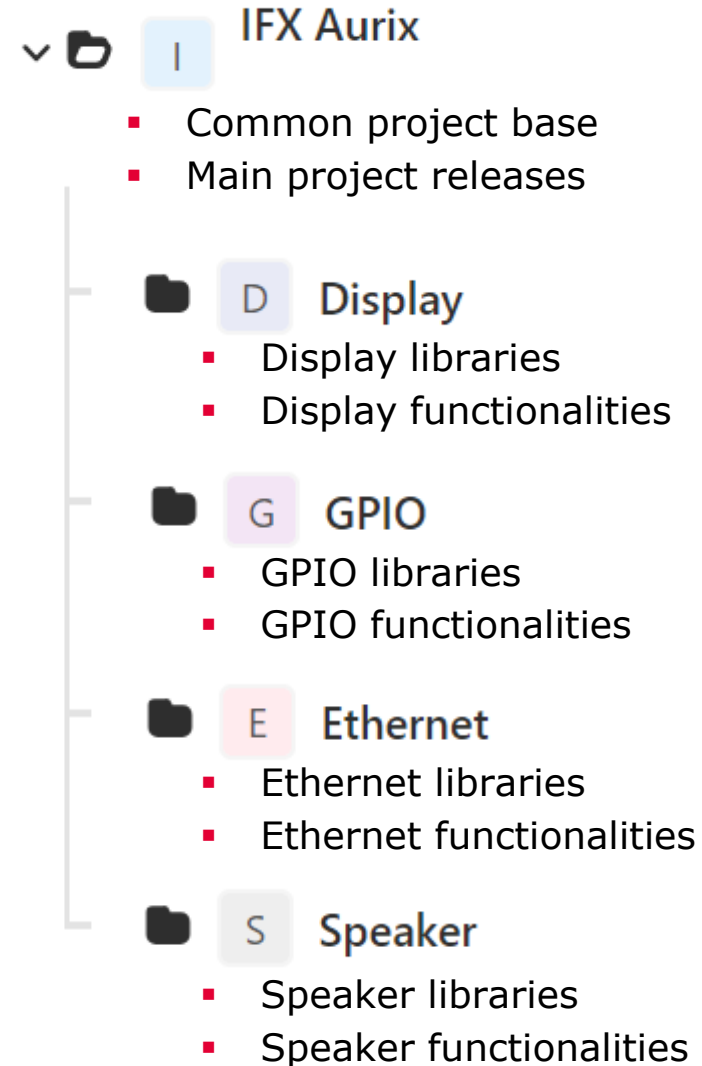


The screenshot shows a GitLab interface for a subgroup. It lists several subgroups with their names and descriptions:

- Course Resources** (C): Projects and resources for the Computer Architectures course
- NXP LandTiger** (N): LANDTIGER board based on NXP LPC1768 SoC embedding ARM Cortex-M3 – Special Projects
- IFX Aurix** (I): AURIX board based on INFINEON TRICORE multiprocessor SoC – Instrumented Bike project
 - Display** (D)
 - GPIO** (G)
 - Ethernet** (E)
 - Speaker** (S)

Main **idea** is to have:

- › a common base (the basic working environment project in the main group)
- › common main developments (projects for main releases in the main group)
- › common set of stable libraries (master branch of the main project in each subgroups) + dev branches
- › parallel development of specific functionalities in the other projects of each subgroup (stable master + dev branches)



GitLab permissions

› Groups

- **Private** The group and its projects can only be viewed by members.
- **Internal** The group and any internal projects can be viewed by any logged in user.
- **Public** The group and any public projects can be viewed without any authentication.

› Projects

- **Private** Project access must be granted explicitly to each user.
- **Internal** The project can be accessed by any logged in user.
- **Public** The project can be accessed without any authentication.

› User permissions

- **Guest** View/pull project, Create/view issues, Leave comments
- **Reporter** Guest + Manage issues, View merge requests
- **Developer** Reporter + Create/manage merge requests, Create new branches, Manage non-protected branches (remove, push on...)
- **Maintainer** Reporter + Manage protected branches (remove, push on, protect/unprotect...), Add new team members, ...
- **Owner** Maintainer + ...