# ECUID Tracking Project

Jason Zhao, LG Electronics

## Goal

Keep track of the unused ECUIDs in some way.

## Motivation

GM limits the amount of ECUIDs to be downloaded by us and other teams, therefore we want to keep track of the usage of ECUID internally due to a potential situation where we deplete all available ECUIDs.

## Background

If the STID folder has a bin file, it means that the unique ECUID in that file is assigned to that unique STID folder. Original ECUIDs in a big chunk and available in storage folder. (One ECUID 9 lines. A STID is 9 digits.)

Some ECUIDs are not assigned yet. A program was used to assign ECUIDs to STID folders. Sarah has all the files.

Global B and GEM are the only ones that use ECUIDs. Reflash team creates STIDs from a range, and they are concerned with lost history when desktop fails. Other teams (Korea, Mexico, etc.) have to send requests to the reflash team to get STID folders with ECUIDs. Also note that Sarah does not delete STID folders but only copies them from storage when they are used by reflash team or other teams, and she records the used ones in multiple sheets in multiple excel files. As reported by Sarah, some of the STID files have their ECU bins moved around so I need to wait until Sarah uploads the local copy on her computer to perform the procedure.

## Procedure

* Get the lists of STIDs used/unused from excel files by program and manually.
  + Check if the lists of STIDs include STID folders that do not contain ECUID
  + STID without CCM and VIM columns are not used yet
  + GEN11\_MASTER\_TCP\_GA\_NEW.xlsx (only GEM): manual inspection
  + GEN11\_MASTER\_TCP\_GB\_NEW.xlsx: use a short python program since it’s hard to keep track of the STIDs manually
  + This step should output a dictionary object for efficiency
    - Key: Value is STID: true/false
* Find out the corresponding used/unused ECUIDs by inspecting Sarah’s STID folders. At the same time, move unused ECUIDs from the STID folders to one big folder.
  + Read previously saved dictionary from above step and iterate over all STID folders
* Final output and modifications we need:
  + A fold that contains all unused ECUIDs
  + A list or folder that tracks the used ECUIDs
  + Clear out ECUIDs in unused STID folders and assign ECUIDs when needed (Ex. send 1000 of such STID folders when depleted)

## Result

* A folder that contains all unused ECUIDs.
* A list of used/unused STIDs and ECUIDs.
* A storage of STID folders that contain used STID folders with an ECUID and unused STID folders without an ECUID.