# Software-Defined Reference-Signal-Independent Implementation of Phase-Locked Loop (PLL): Initialization using AI Model

Dr. Christoph Q Lauter, C. Alejandra Carreon-Jimenez, Gabriel Miki-Silva, et al.

This project is part of a broader initiative focused on designing a software-defined, GPS-independent Phase-Locked Loop (PLL) system for use in telecommunications environments where GPS synchronization is either unreliable or unavailable. Our implementation is based on the DAHDI (Digium Asterisk Hardware Device Interface) timing tool, and specifically aims to modify its internal software timer using a programmable field referred to as `fudge`.

The core objective of our work is to enhance the reliability of signal synchronization in remote or underserved regions. To achieve this, our team has carried out several foundational tasks: configuring and compiling DAHDI from the Osmocom repository, modifying its kernel timer functions to include the `fudge` parameter, and implementing custom `ioctl(driver-styled)` calls to access this functionality from user space. Additionally, user-level tools such as `dahdi_fudge` have been developed to test and adjust the timer behavior, allowing for dynamic experimentation.

An important aspect of this project is the integration with `osmo-e1d`, an E1 signal handler, which monitors FIFO queue lengths and dynamically adjusts the `fudge` value to maintain synchronization. This integration provides a feedback loop between hardware-level signal timing and software-defined corrections, enabling adaptive behavior as conditions fluctuate.

To improve initialization—specifically determining the optimal value for the first `fudge` iteration—we are developing a machine learning model in collaboration with Dr. Talukder's lab. This model will be trained on two distinct latency datasets: one generated from raw, non-GPS-backed signals (representing unstable conditions), and another from stable, GPS-backed signals (representing ideal output). The model's purpose is to infer an optimal starting fudge value that can enhance synchronization accuracy from the very beginning of execution.

Our approach blends low-level systems engineering, data-driven learning techniques, and practical tool development in an effort to make synchronization tools more resilient and accessible. The project is designed to serve real-world needs, particularly for decentralized or under-resourced regions where GPS infrastructure may be inconsistent or entirely absent.

**Conclusion:** While still in its early stages, this project has already established a technically sound and versatile platform. By combining programmable timing infrastructure with AI-guided initialization, we aim to produce a lightweight yet intelligent PLL implementation suitable for infrastructure-constrained environments. Future work will explore broader training datasets, refined fudge control loops, and real-world deployment scenarios. This is a collaborative effort, and we welcome further contributions, insight, and experimentation.