

Πανεπιστήμιο Θεσσαλίας - Τμήμα Ηλεκτρολόγων Μηχανικών και
Μηχανικών Υπολογιστών

ΗΥ439 - Αλγόριθμοι CAD II - Φυσικής Σχεδίασης

Εαρινό Εξάμηνο - Ακαδημαϊκό Έτος 2023-2024

2η Εργασία

7/4/2025 έως 20/4/2025

X. Σωτηρίου

2η Εργασία

Οι στόχοι της 2ης εργασίας είναι (1) να χρησιμοποιήσετε το παρεχόμενο API ώστε να φορτώσετε στην στο εργαλείο σας αρχεία LIB, LEF, DEF και VERILOG, καθώς και (2) να απεικονίσετε τα στοιχεία του κυκλώματος στην οθόνη. Η τρέχουσα μορφή του API είναι η παρακάτω, προς το παρόν χρησιμοποιούμε C, στην 3η εργασία θα πάμε σε C++.

```
// API - file api.h for libpathviz.so //

// *** Function Prototypes *** //

// General Functions //
extern void init_globals();

// Placement Related //

extern unsigned long get_total_components(int);
extern unsigned long get_total_placed_components(int);
extern int get_component_iterator(unsigned long *, int *);
extern int next_component_iterator(unsigned long *, int *);
extern void get_component_name(unsigned long, int, char **);
extern int get_component_is_placed(unsigned long, int);
extern int get_component_location(unsigned long, int, double *, double *);

// TCL Commands Interface //

int load_lef(ClientData, Tcl_Interp *, int, Tcl_Obj *const*);
int load_verilog(ClientData, Tcl_Interp *, int, Tcl_Obj *const*);
int load_def(ClientData, Tcl_Interp *, int, Tcl_Obj *const*);

// *** Externals *** //

extern int currentflowstep;
extern int lefloaded;
extern int libloaded;
extern int defloaded[];
extern int verilogloaded[];
```

Figure 1: api.h

Θα πρέπει να κάνετε compile με την shared βιβλιοθήκη `libpathviz.so` που θα κατεβάσετε από την σελίδα του μαθήματος. Σαν σημείο αναφοράς μπορείτε να κατεβάσετε και το εργαλείο από το GitHub (κλειστού κώδικα) για να δοκιμάσετε την διαδικασία και να συγκρίνετε το layout. Το σχετικό URL είναι: <https://github.com/Silicon-Highway-Technologies/Si-Time>. Παρακάτω φαίνεται πως θα πρέπει να κάνετε compile το πρόγραμμά σας.

```
export LD_LIBRARY_PATH=.
```

```
gcc -o <output> <YOUR C FILES> -g 'pkg-config --cflags gtk+-2.0'
-lm 'pkg-config --libs gtk+-2.0' -lgthread-2.0 -pthread -lglib-2.0
-lreadline -ltcl -L./ -lpathviz
```

Figure 2: compilation script

Αφού φορτώσετε LEF, DEF ή LEF, VERILOG, DEF (δεν χρειάζεται να φορτώσετε LIB σε αυτή την εργασία), θα πρέπει να ζωγραφίσετε την τοποθέτηση του κυκλώματος χρησιμοποιώντας το GUI που υλοποιήσατε στην 1η Εργασία. Το πιο βολικό είναι να χρησιμοποιήσετε ένα TCL front-end από το οποίο θα καλείτε τις συναρτήσεις του API. Παρακάτω φαίνεται ένα τέτοιο παράδειγμα.

```
// *** load_lef_TCL *** //
// command: load_lef //
int load_lef_TCL(ClientData clientdata, Tcl_Interp *interp, int objc, Tcl_Obj *const* objv)
{
    load_lef(clientdata, interp, objc, objv);
}
```

Figure 3: TCL API function definition Example

Μπορείτε να ζωγραφίσετε απευθείας το layout, ή όταν ο χρήστης χρησιμοποιήσει κάποια εντολή TCL.

Παρουσίαση καί Demo

Προετοιμάστε ένα σύντομο ppt και μια επίδειξη του εκτελέσιμου για την ώρα του μαθήματος.

Η προθεσμία παράδοσης του 1ου Συνόλου Ασκήσεων είναι η **20/4/2025**. Μέχρι τότε θα πρέπει να έχετε υποβάλλει τις λύσεις των ασκήσεων μέσω του e-Class.