rEDActor – A PDK Cross-platform Integrated Development Environment for Semiconductor Technologies

Datsuk Anton

MES2020, Russia

_

Motivation

	rEDActor	Nedit	Sublime Text	SKILL IDE
Integration with EDA software (Virtuoso, ADS)	✔️	❌	❌	✔️
Code auto-formating	✔️	❌	❌	❌
Code folding	✔️	❌	✔️	✔️
Keyword auto-completion	✔️	❌	✔️	✔️
Embedded syntax checker	✔️	❌	❌	✔️
Auto initialization of global variables	✔️	❌	❌	❌
Code Debugging	✔️	❌	❌	✔️
SKILL to AEL code conversion	✔️	❌	❌	❌

Integration with leading edge EDA vendors

Code execution
from rEDActor environment
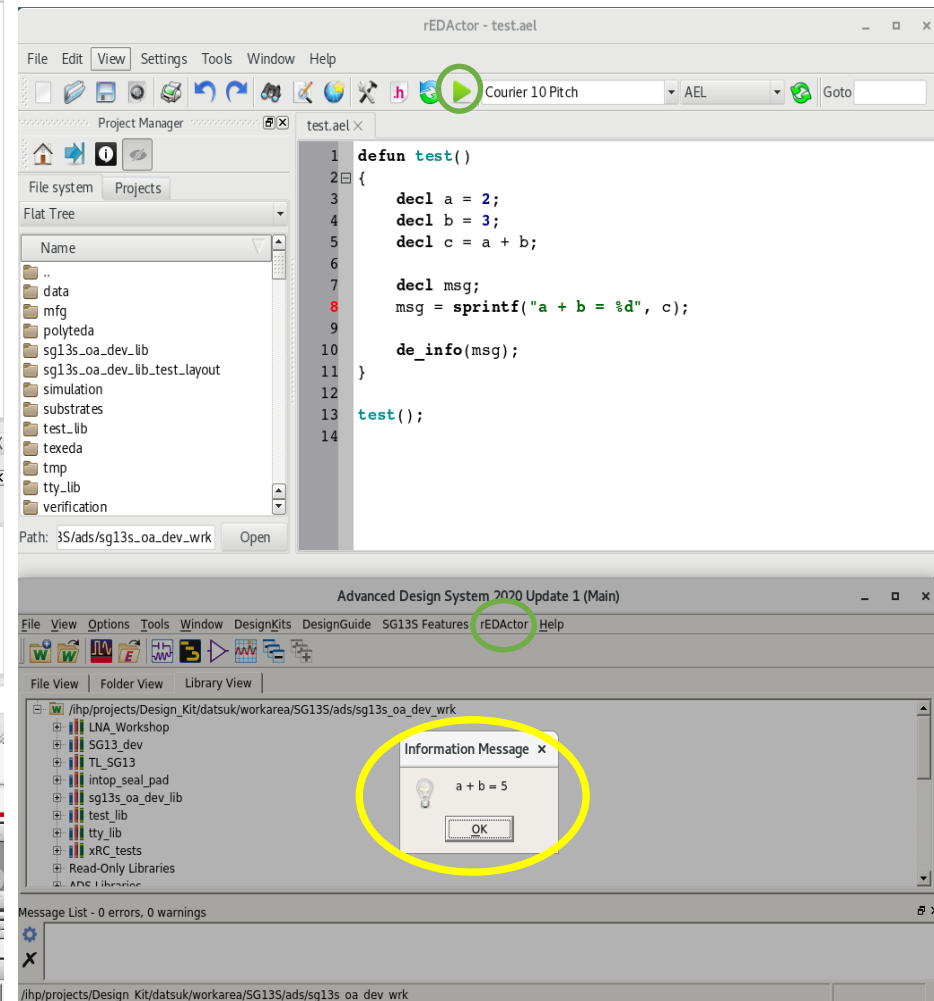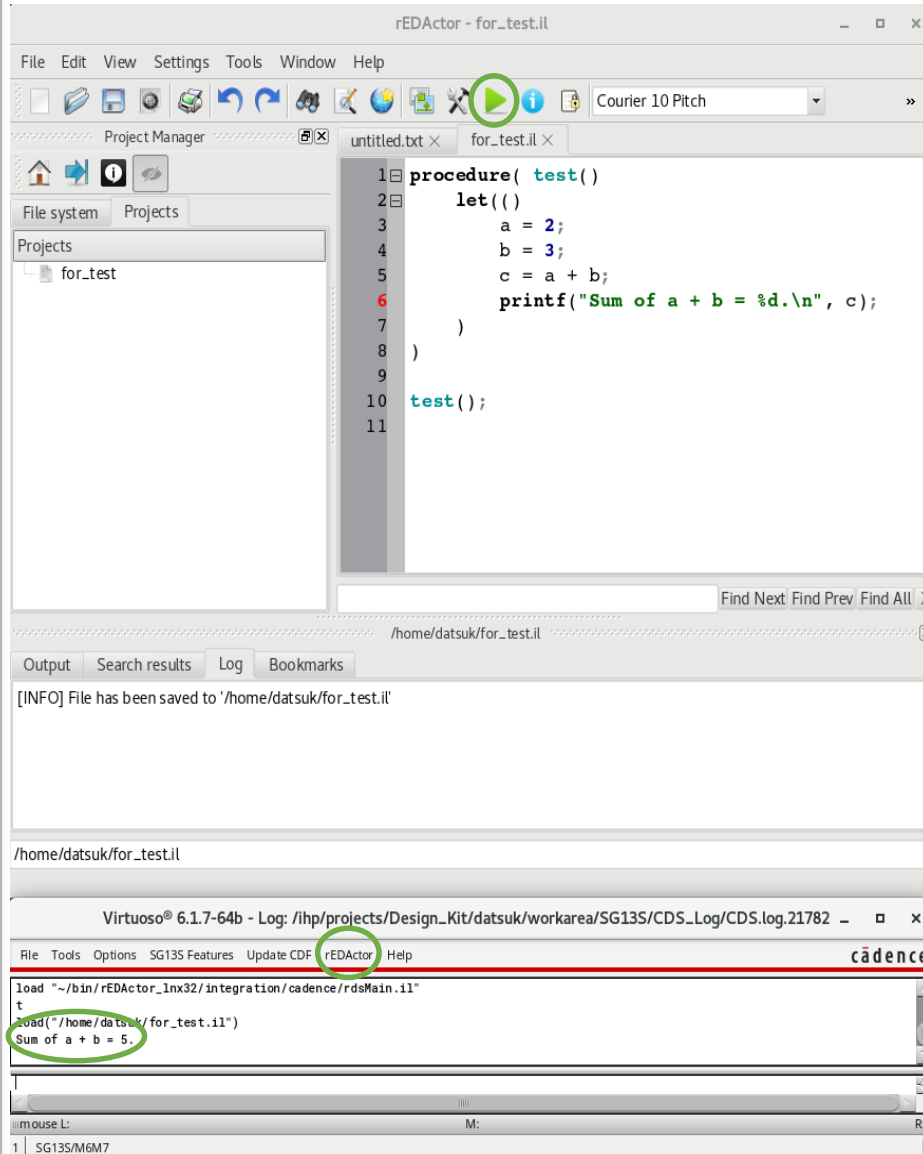
Code autoformating
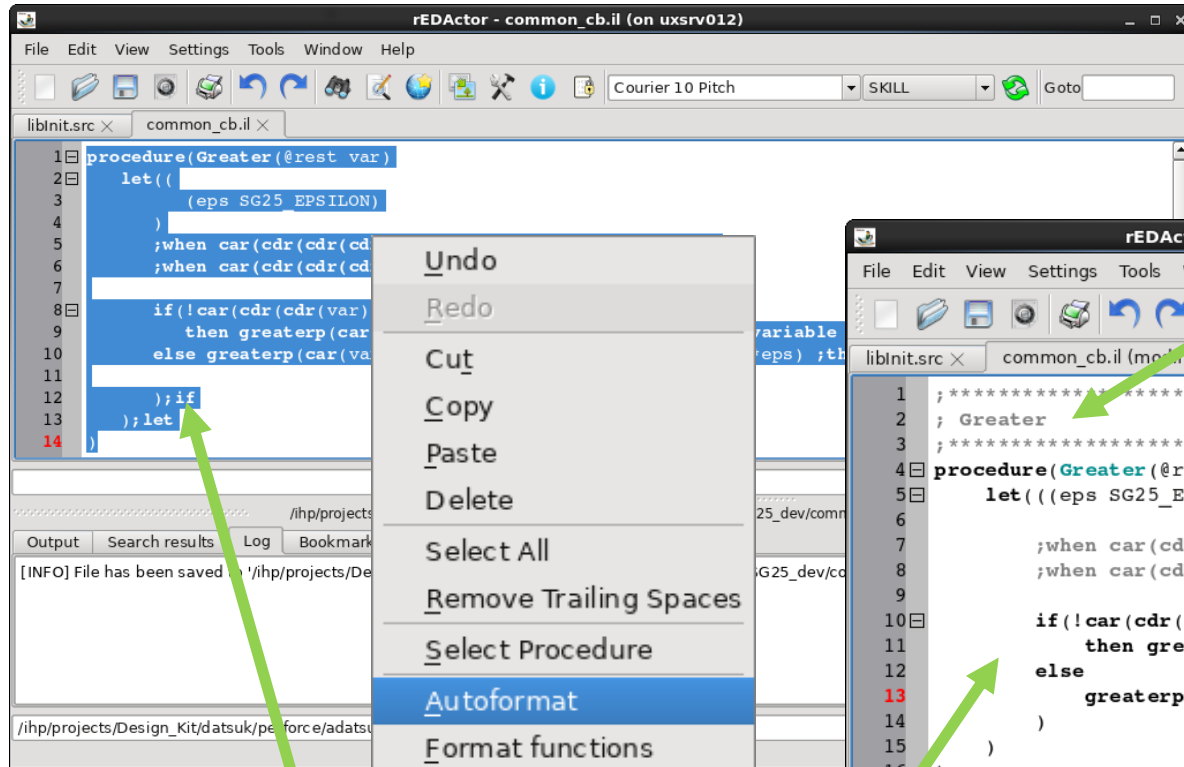
Adds footer

Removes unrequired comments (on demand)
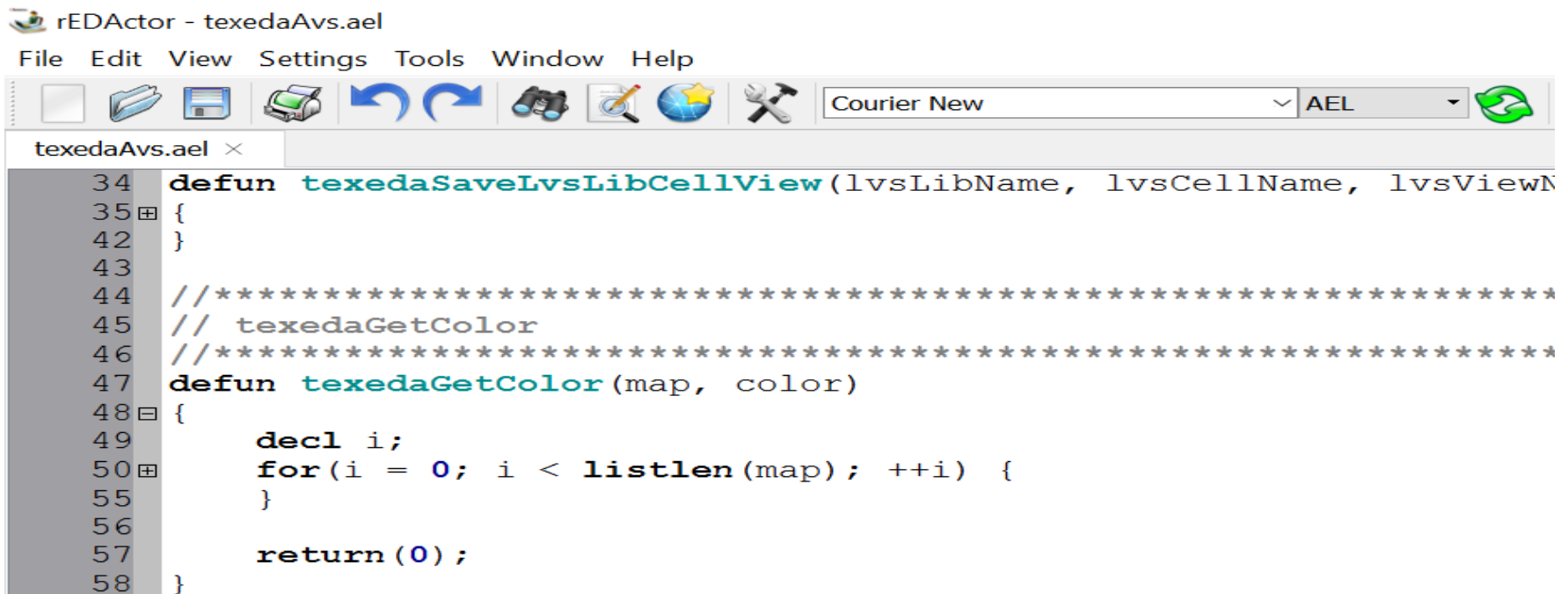
Reformats the code in accordance to user settings
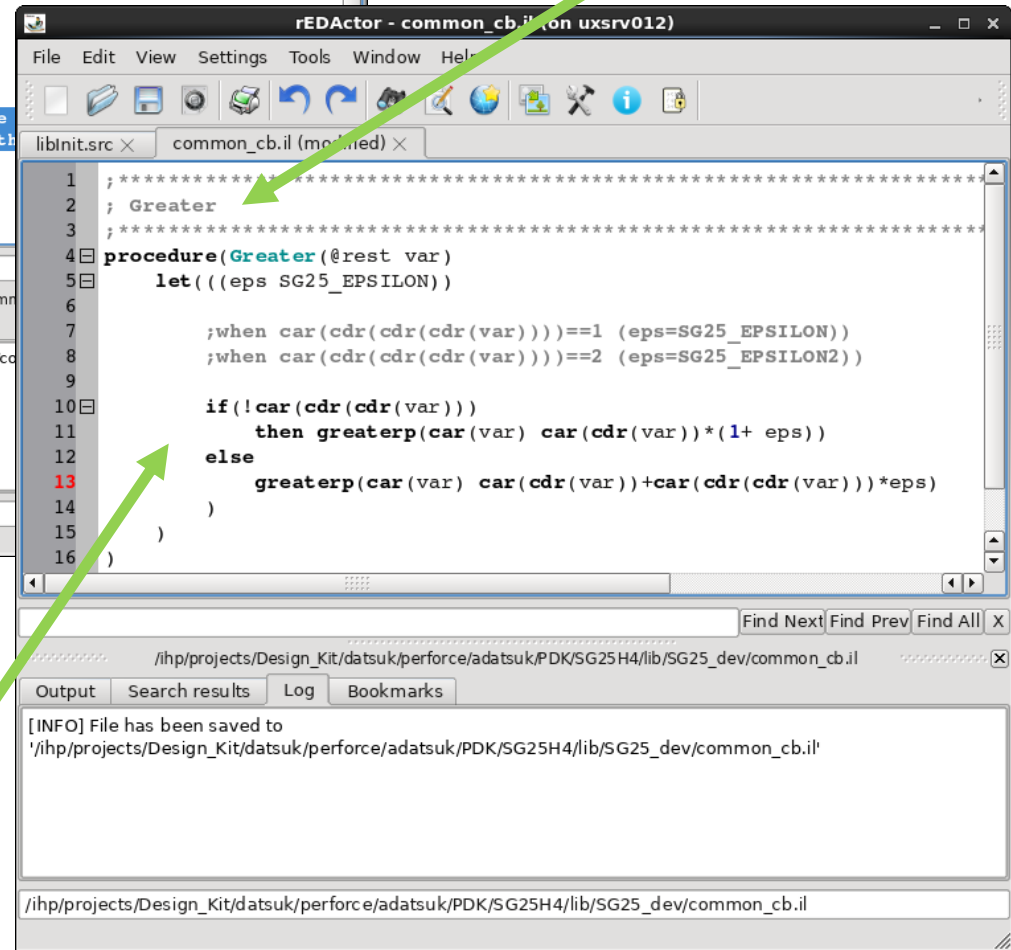
Code folding
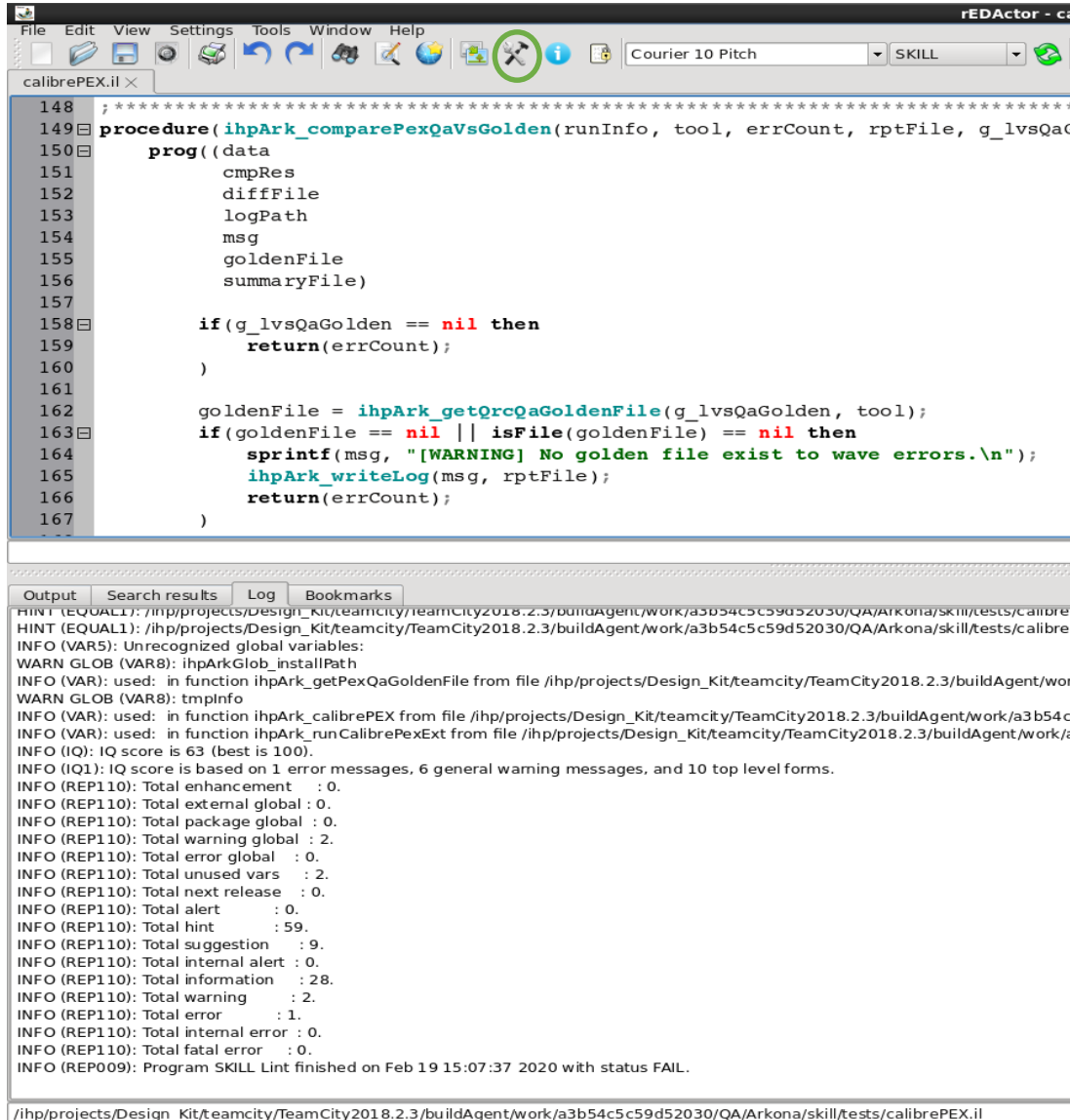
- IDE rEDActor allows the user to selectively hide and display (fold) sections of a currently edited file.

- This simplifies managing large amounts of text while viewing only those subsections of the text that are specifically relevant at any given time.

Embedded syntax checkers

rEDActor supports syntax checkers for:
- SKILL (Cadence)
- AEL (Keysight)
- SVRF (Mentor)
- LAYTOOLS (TexEDA)
- XML .
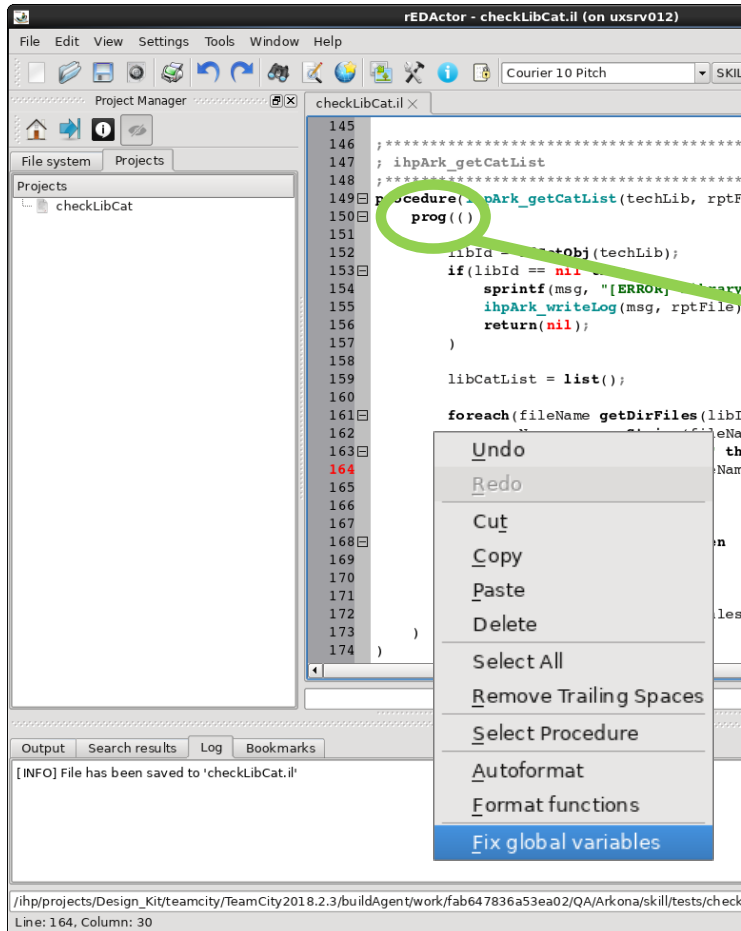
On clicking on the line number when the error was found rEDActor automatically navigates to this line.

Auto initialization of global variables

IDE finds the global variables and brings them under procedure scopes.

Global variables decrease flexibility of the program and lead to errors.

Code debugging
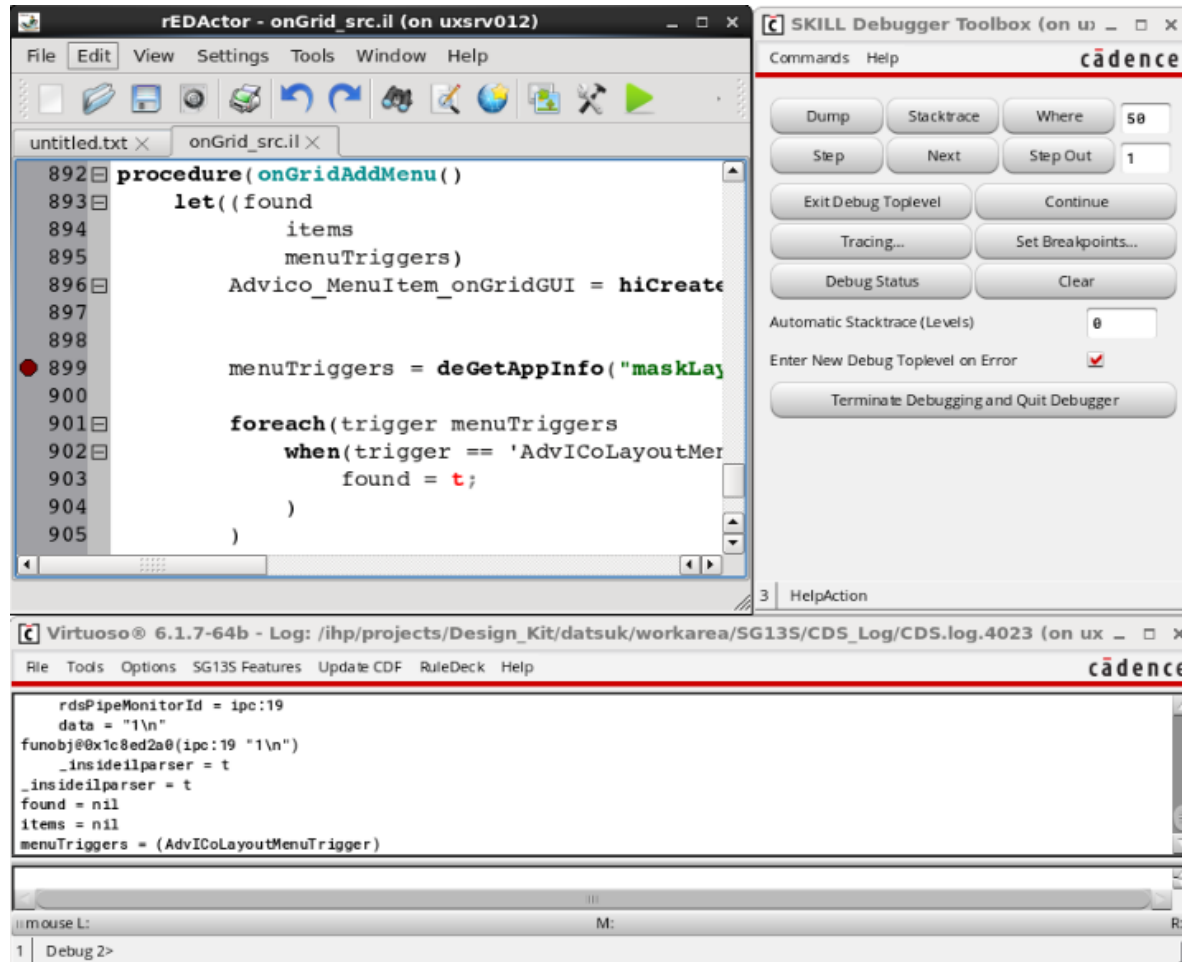
If required a developer can set up break points and perform SKILL code debugging.

The debugger allows to output information for:
- Global and local variables
- Functions stack calls

SKILL to AEL code conversion

<div align="center">

SKILL ➡ AEL

</div>

```skill
when( metallization=="M4" || metallization=="M4M5"
  bmlayer="Metal3")

; check minvalues
when(w < wmin-epsilon)
  w=wmin;
  hiGetAttention()
  printf("Width < %L\n" wmin)
)
when(l < lmin-epsilon)
  l=lmin;
  hiGetAttention()
  printf("Height < %L\n" lmin)
)

; how many vias?
xanz=fix((w-cont_over-cont_over+cont_dist)/(cont_size+cont_dist)+epsilon)
; width for vias
w1=xanz*(cont_size+cont_dist)-cont_dist+cont_over+cont_over
; offset to first via
xoffset=(w-w1)/2
xoffset=GridFix(xoffset)

yanz=fix((l-cont_over-cont_over+cont_dist)/(cont_size+cont_dist)+epsilon)
l1=yanz*(cont_size+cont_dist)-cont_dist+cont_over+cont_over
yoffset=(l-l1)/2
yoffset=GridFix(yoffset)

ycont_cnt=cont_over+yoffset
; draw vias
while(ycont_cnt+cont_size+cont_over <= l+epsilon
 xcont_cnt=cont_over+xoffset
 while(xcont_cnt+cont_size+cont_over <= w+epsilon
  dbCreateRect(pcCellView vialayer list(
   list(xcont_cnt ycont_cnt)
   list(xcont_cnt+cont_size ycont_cnt+cont_size))
  )
  xcont_cnt=xcont_cnt+cont_size+cont_dist
 )
 ycont_cnt=ycont_cnt+cont_size+cont_dist
)
```

```ael
if(metallization == "M4" || metallization == "M4M5") {
    bmlayer = "Metal3";
}
if(w < wmin-epsilon) {
    w = wmin;
    hiGetAttention();
    de_info(sprintf("Width < %L\n", wmin));
}
if(l < lmin-epsilon) {
    l = lmin;
    hiGetAttention();
    de_info(sprintf("Height < %L\n", lmin));
}
xanz = fix((w-cont_over-cont_over+cont_dist)/(cont_size+cont_dist)+epsilon);

w1 = xanz*(cont_size+cont_dist)-cont_dist+cont_over+cont_over;

xoffset = (w-w1)/2;
xoffset = GridFix(xoffset);

yanz = fix((l-cont_over-cont_over+cont_dist)/(cont_size+cont_dist)+epsilon);
l1 = yanz*(cont_size+cont_dist)-cont_dist+cont_over+cont_over;
yoffset = (l-l1)/2;
yoffset = GridFix(yoffset);

ycont_cnt = cont_over+yoffset;

while(ycont_cnt+cont_size+cont_over<= l+epsilon) {
    xcont_cnt = cont_over+xoffset;
    while(xcont_cnt+cont_size+cont_over<= w+epsilon) {
        dbCreateRect(pcCellView, vialayer,
                     list(xcont_cnt, ycont_cnt,
                     xcont_cnt+cont_size, ycont_cnt+cont_size));
        xcont_cnt = xcont_cnt+cont_size+cont_dist;
    }
    ycont_cnt = ycont_cnt+cont_size+cont_dist;
}
```
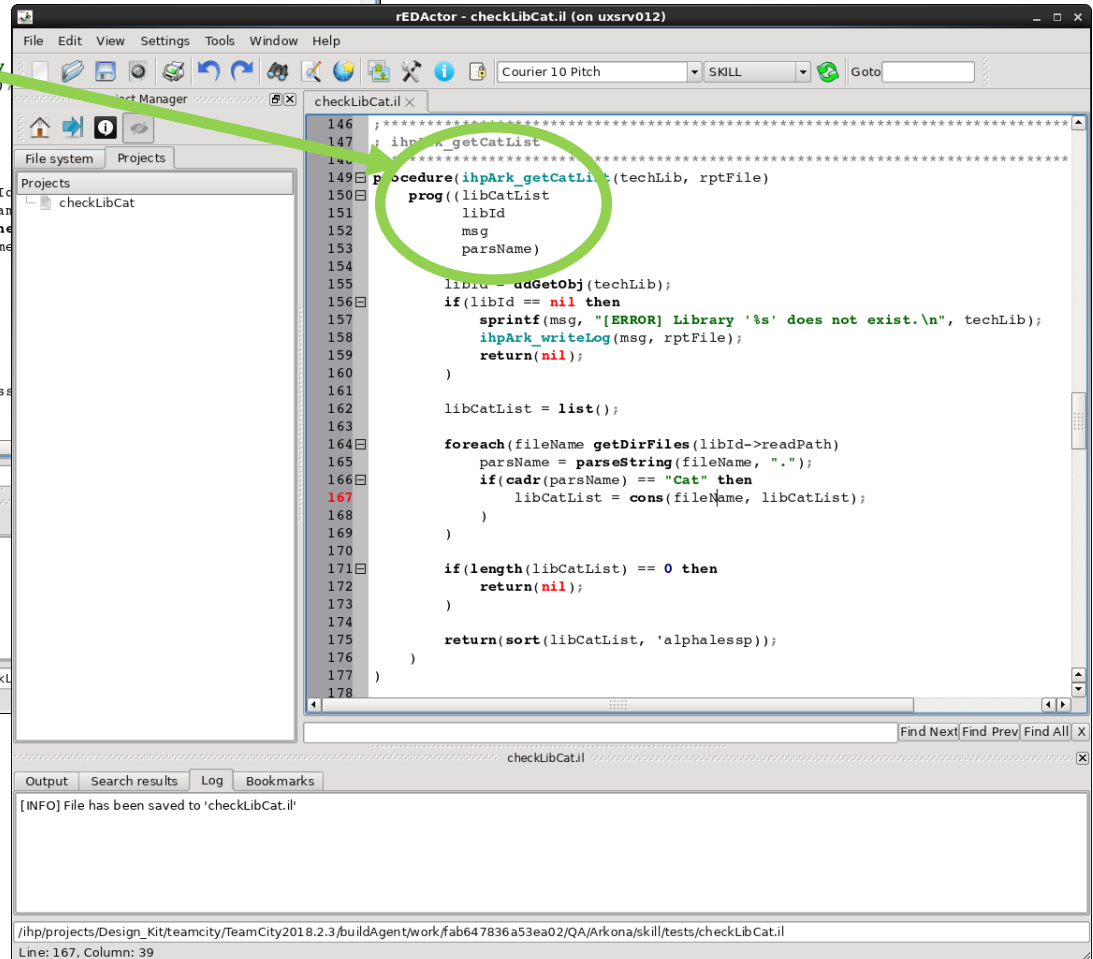
SKILL to AEL code conversion

SKILL

AEL
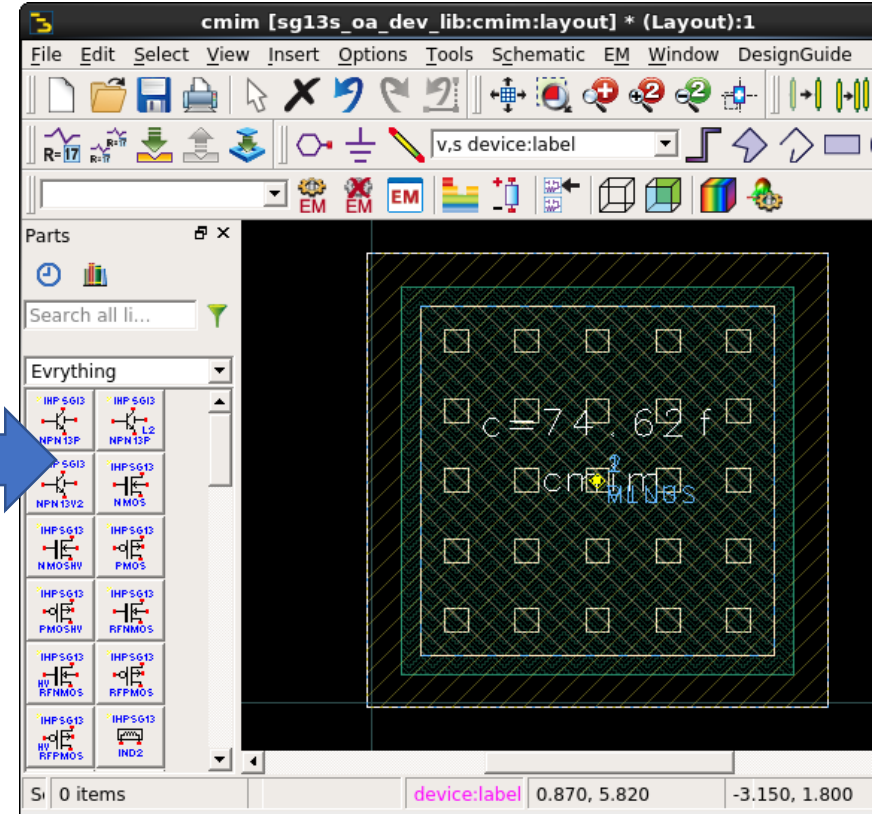
The converted AEL code for CMIM Pcell has been loaded into Keysight ADS environment without any additional modifications.

SKILL to AEL code conversion

The conversion works based on plug-ins.

For this the basic SKILL procedures (for ex. dbCreateRect, dbCreatePolygon, etc.), have been developed using Keysight AEL language.

User can extend the plug-in file on demand.

```
File  Edit  View  Settings  Tools  Window  Help

                                        Courier 10 Pitch        AEL

cadence.ael ×

 1   defun dbCreateRect(g_cxt, g_lpp, g_bbox)
 2   {
 3       decl layerId = db_get_layerid(g_cxt, nth(0, g_lpp),
 4                                            nth(1, g_lpp));
 5       db_add_rectangle(g_cxt, layerId, nth(0, g_bbox),
 6                                        nth(1, g_bbox),
 7                                        nth(2, g_bbox),
 8                                        nth(3, g_bbox));
 9   }
10
11   defun dbCreatePolygon(cxt, layer, points)
12   {
13       db_add_polygon(cxt);
14
15       if(is_string(layer)) {
16           layer = db_get_layerid(cxt, layer, "drawing");
17       }
18       elseif(is_list(layer)) {
19           layer = db_get_layerid(cxt, nth(0, layer), nth(1, layer));
20       }
21
22       decl i;
23       for(i = 0; i < listlen(points); ++i) {
24           decl x = nth(i, points);
25           ++i;
26           decl y = nth(i, points);
27
28           db_add_point(cxt, x, y);
29       }
30
31       decl polygon = db_end(cxt, layer);
32
33       return(polygon);
34   }
```
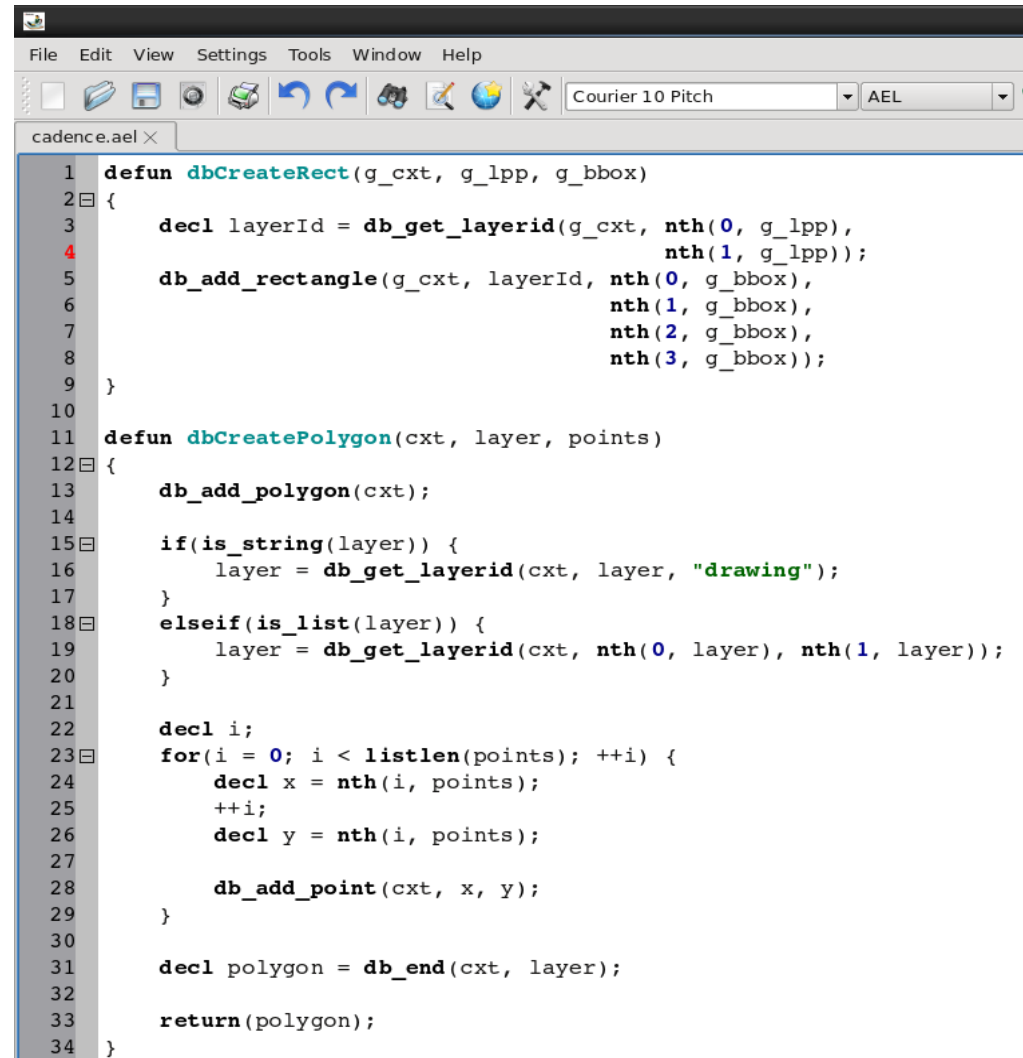
The following companies are using IDE rEDActor:

Conclusions and results

1. rEDActor has been incorporated into the leading-edge software tools increasing efficiency of PDK development.

2. The IDE provides a functionality to perform code autoformating, folding, debugging etc.

3. The editor allows to convert SKILL code to AEL thus increasing productivity of Cadence interoperable PDKs.

4. The IDE rEDActor has been successfully applied for development of PDKs for 0.13 um and 0.25 um SiGe BiCMOS technologies resulting in highly reliable and robust PDK releases.

Thank you for attention!

IHP – Leibniz-Institut für innovative Mikroelektronik
Im Technologiepark 25
15236 Frankfurt (Oder)

Phone: +49 (0) 335 5625 647
Fax: +49 (0) 335 5625 327
E-Mail: datsuk@ihp-microelectronics.com

www.ihp-microelectronics.com