

# کتابچه راهنمای نرم افزار فنی - مهندسی

**DIgSILENT PowerFactory**  
**Version 13.0**

---

شرکت برق منطقه ای فارس

معاونت برنامه ریزی و تحقیقات

۱۳۸۴

شرکت برق منطقه ای فارس

تمیبه شده در :

شیراز- فیابان زند- نبش فیابان فلسطین

شرکت برق منطقه ای فارس

تلفن : ۰۷۱۱-۲۳۳۰۰۳۱-۹

فاکس : ۰۷۱۱-۲۳۵۹۰۴۷

[www.frec.co.ir](http://www.frec.co.ir)

وزارت نیرو

تماس با مترجمان

مهمربنا کلساز شیرازی [mshirazi@frec.co.ir](mailto:mshirazi@frec.co.ir)

امیر فرشیان فسایی [farshchian@frec.co.ir](mailto:farshchian@frec.co.ir)

### حق چاپ و انتشار انحصاری

تمامی این ترجمه در شرکت برق منطقه ای فارس و با همکاری کارشناسان دفتر برنامه ریزی فنی و برآورد بار معاونت برنامه ریزی و تحقیقات تهیه شده است. بنابراین کلیه حقوق این ترجمه متعلق به شرکت برق منطقه ای فارس بوده و هرگونه نسخه برداری بدون کسب اجازه از این شرکت، ممنوع بوده و پیگرد قانونی دارد.

تابستان ۱۳۸۴ - شیراز  
شرکت برق منطقه ای فارس

وزارت نیرو




**Basic  
User's  
Manual**

شرکت برق منطقه ای فارس

## فهرست مطالب

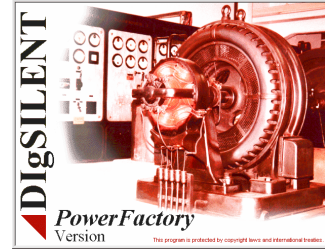
<b>1</b>	<b>Introduction.....</b>	<b>7</b>
1.1	Changes from Earlier Versions .....	8
1.2	Database Management .....	9
1.3	DIgSILENT "Heavy" and "Lightweight" Versions .....	10
1.4	DIgSILENT Help System .....	10
1.5	The Tutorial Manager .....	11
1.6	Conventions Used In This Manual .....	12
1.7	DIgSILENT PowerFactory Windows .....	13
<b>2</b>	<b>Logon, User Accounts and Program Settings</b>	<b>17</b>
2.1	Program Configurations and Settings .....	17
2.2	License Settings .....	19
2.3	The User Account Managing System .....	21
2.4	Users and User Groups .....	24
<b>3</b>	<b>Toolbar Definition</b>	<b>28</b>
<b>4</b>	<b>User Settings</b>	<b>35</b>
4.1	General Settings.....	35
4.2	Graphic Windows Settings.....	36
4.3	Data Manager Settings .....	37
4.4	Output Window Settings .....	37
4.5	Functions Settings.....	38
4.6	Directories .....	38
<b>5</b>	<b>The Database and the Data Manager</b>	<b>39</b>
5.1	Data Base Contents.....	40
5.2	Using the Data Manager .....	41
5.3	Moving Around in the Database Tree.....	43
5.4	Adding a New Item.....	44
5.5	Deleting an Item .....	46
5.6	Cut, Copy, Paste and Move Objects .....	47
5.7	Searching Items .....	49
5.8	Editing Data Objects .....	51
5.9	Additional Features.....	55
5.10	The Flexible Data Page Tab .....	57
5.11	Monitor Variable Sets .....	58
5.12	Save and Restore Parts of the Database .....	60
5.13	Spreadsheet Format Data Import/Export .....	60
5.14	Converting/Importing PSS/E Files .....	66
5.15	Importing Version 3.1x Files.....	71
5.16	The Input Window .....	73
5.17	Input Window Commands .....	74

5.18	The Database Manager Message Bar.....	76
5.19	Special Folders in the Database.....	76
5.20	Special Objects in the Database.....	79
<b>6</b>	<b>Project Management .....</b>	<b>83</b>
6.1	Creating a new Project .....	85
6.2	The Project Dialog .....	86
6.3	Projects and Study Cases .....	90
6.4	Projects and System Stages .....	94
6.5	Reducing a System Stage .....	98
6.6	Comparing Projects.....	100
<b>7</b>	<b>Graphic Windows.....</b>	<b>101</b>
7.1	The Page Tab.....	101
7.2	The Page Tab Menu .....	101
7.3	New Graphic Window .....	102
7.4	Linkage Between Graphics and Database.....	103
7.5	Building from Predefined Objects .....	105
7.6	Using the Graphic Windows .....	108
7.7	Graphic Layers .....	112
7.8	Basic Graphic Functions .....	114
7.9	Single Line Graphics.....	123
7.10	Interconnecting Power Subsystems .....	125
<b>8</b>	<b>Power System Definition .....</b>	<b>132</b>
8.1	Station and Busbar Systems .....	132
8.2	Transmission Lines and Cables .....	141
8.3	Characteristics, Scales and Triggers .....	144
8.4	Handling Scales and Characteristics.....	156
8.5	The Feeder object .....	159
<b>9</b>	<b>Protection.....</b>	<b>162</b>
9.1	Creating a Protection Device.....	162
9.2	Basic Protection Devices.....	172
9.3	Short-Circuit Sweep.....	189
9.4	The Time-Distance Diagram .....	191
9.5	Path Definitions .....	196
9.6	Protection Analysis Results.....	197
9.7	Time-Overcurrent Plots.....	198
9.8	The Curve-Input Command .....	206
<b>10</b>	<b>Performing Calculations.....</b>	<b>209</b>
10.1	Results .....	211
10.2	Load Flow Calculations .....	211
10.3	Short-Circuit Calculations .....	215
10.4	Harmonics Analysis .....	219
10.5	Modal Analysis / Eigenvalue Calculation.....	230
10.6	Comparisons Between Calculations .....	234
<b>11</b>	<b>Virtual Instruments.....</b>	<b>237</b>

11.1	Result Graphs .....	243
11.2	Labeling Plots .....	245
11.3	Subplots.....	250
11.4	The Curve Filter.....	256
11.5	The Voltage Profile Plot.....	257
11.6	The Waveform Plot.....	259
11.7	The Vector Diagram .....	262
11.8	Embedded Graphic Windows.....	264
<b>12</b>	<b>Results, Graphs and Documentation</b>	<b>266</b>
12.1	Result Objects.....	266
12.2	The Output Window .....	268
12.3	The Form Editor .....	273
12.4	The DIgSILENT Output Language .....	276
12.5	The Output of Device Data .....	283
12.6	Output of Results .....	287
12.7	The Annex for Documentation.....	288
<b>13</b>	<b>DOLE and On-Line Data Exchange</b>	<b>289</b>
13.1	DOLE Interface .....	289
13.2	DIgSILENT On-Line Operation Mode.....	301
<b>14</b>	<b>Conversion from Graphical Information Systems</b>	<b>306</b>
14.1	".DGS"-File Creation by using the Excel Sheet.....	306
14.2	The Power Factory Import .....	307
14.3	". DGS" - File Format .....	309
 <b>15</b>	<b>DPL</b>	<b>311</b>
15.1	The DPL Command Object.....	312
15.2	The DPL Script Language.....	313
15.3	Access to Other Objects .....	317
15.4	Access to Locally Stored Objects .....	319
15.5	Accessing the General Selection .....	320
15.6	Accessing External Objects.....	321
15.7	Remote Scripts and DPL command Libraies .....	322
15.8	DPL Functions and Subroutines .....	324
<b>16</b>	<b>Frequently Asked Questions .....</b>	<b>328</b>

## Chapter 15

# DPL



زبان برنامه نویسی *DIgSILENT* یا "DPL" از طریق ایجاد توابع محاسباتی جدید، بعد جدیدی را به برنامه افزوده است. نمونه هایی از این توابع عبارتند از :

- بهینه سازی حلقه های باز
- بهینه سازی توسعه سیستم
- هماهنگی حفاظتی اتوماتیک
- آنالیز پایداری ولتاژ خاص
- ارزیابی ظرفیت انتقالی
- محاسبات هزینه های انتقال

این توابع محاسباتی جدید، در قالب برنامه های اسکریپت نوشته می شوند که در برگیرنده :

- دستورات حلقه ای مانند 'if-then-else' و 'do-while'
- دستورات *DIgSILENT* (مانند دستورات پخش بار یا اتصال کوتاه)
- روالهای ورودی و خروجی
- عبارتهای ریاضی
- روشهای فراخوانی های اجزاء *DIgSILENT*
- فراخوانی برنامه های فرعی

این اسکریپت ها چنین به وسیله اجزای دستور DPL اجرا می شوند.

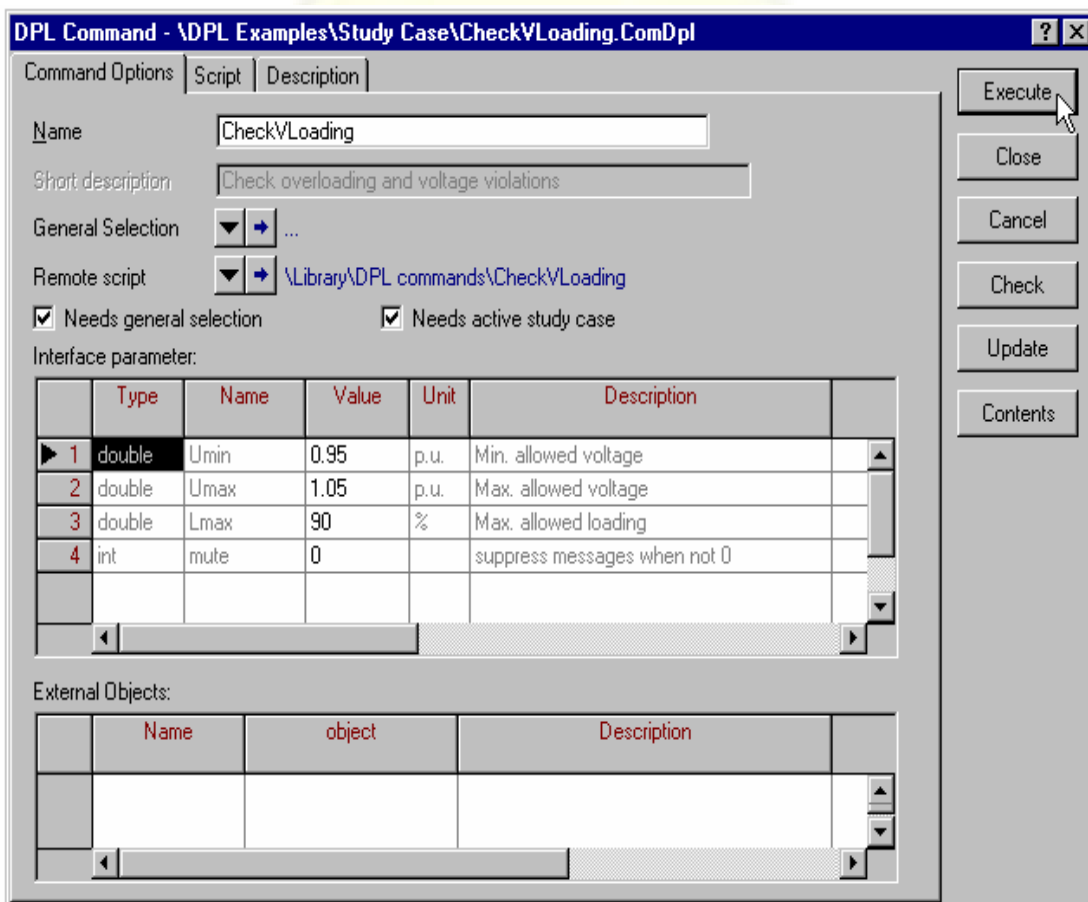
برای درک فلسفه DPL و ساختار سلسه مراتبی نتیجه اسکریپت های DPL درک موارد زیر الزامی است :

- یک فرمان DPL یا اسکریپت خودش یا اسکریپت های دیگر فرامین DPL را اجرا می کند. در حالت اول، فرمان DPL را **'root command'** و اسکریپت مربوطه، **'local script'** نامیده می شود. در حالت دوم، فرمان DPL را **'referring command'** و اسکریپت مربوطه، **'remote script'** نامیده می شود.
- **'root command'** می تواند متغیرهای واسط کاربر را که از بیرون اسکریپت قابل دستیابی بوده به عنوان مقادیر پیش فرض، تعریف نماید.
- هر **'root command'** می تواند یک یا چند عنصر خارجی را تعریف کند. عناصر خارجی به منظور اجرای فرمان DPL با عناصر شبکه قدرت مورد نظر یا منتخبی از آن، استفاده می شوند.
- یک **'referring command'** می تواند باعث کنار گذاشتن تمام مقادیر رابط پیش فرض و همه عناصر خارجی انتخاب شده از فرمان **'remote script'** گردد.
- هر فرمان DPL می تواند در قالب یک زیربرنامه، بوسیله دیگر فرمانهای DPL فراخوانده شود.

استفاده از نسخه های راه دور، موضوعات خارجی و متغیرهای رابط امکان ایجاد فرامین کلی DPL را که می توانند با تنظیمات مختلف در بیشتر پروژه ها و حالت های مطالعه استفاده شوند را امکان پذیر می سازند.

### 15.1 The DPL Command Object

موضوع فرمان DPL از یک مرجع برای فرمان DPL راه دور نگهداری می کند البته وقتی که آن یک فرمان ریشه نباشد. مثال شرح داده شده در شکل ۱۵-۱ ظاهرا یک فرمان ارجاعی است، از آنجایی که مرجع "DPL SCRIPT" در دسته فرمان های راه دور (remote). (remote). "Library \ DPL Commands \ CheckVLoading" می باشد. یک فرمان ریشه دارای نسخه خود در صفحه "Script" توضیح داده شده است. فرمان ارجاعی از نسخه این فرمان DPL راه دور استفاده می کند.



شکل ۱۵-۱: فرمان DPL

هم چنین فرمان DPL در یک مرجع انتخاب موضوعات "General Selection" نگهداری می شود. این انتخاب کلی در این مثال خالی است. فقط یک "General Selection" تک در یک زمان برای همه نسخه های DPL معتبر است. این جمله بدین معناست که تنظیم "General Selection" در کادر محاوره ای فرمان DPL "General Selection" (انتخاب کلی) برای همه فرامین در این پایگاه داده ها را تغییر می دهد.

این بخش رابط، برای تعیین متغیرهایی که از بیرون خود فرمان DPL قابل دستیابی هستند، استفاده می‌شود. فرامین DPL که دیگر فرامین آن مانند زیر روالها را فرا می‌خوانند را می‌توان برای تغییر اندازه های متغیرهای رابط این زیر روالهای DPL استفاده کرد.

این فهرست موضوعات خارجی برای اجرای فرمان DPL در مورد موضوعات خاص استفاده می‌شود. یکی از فرمانها DPL، برای مثال، یک مجموعه از خطوطی که اتصال کوتاه باعث افت ولتاژ در باسبارهای معین گردیده، جستجو کرده که به آن باس بار خاص به عنوان یک موضوع خارجی دسترسی پیدا می‌کند. البته مهمترین قسمت یک فرمان ریشه DPL، بطور مسلم نسخه برنامه DPL واقعی می‌باشد. آن نسخه روی صفحه "Script" از کادر محاوره ای فرمان ریشه DPL نوشته می‌شود.

## 15.2 The DPL Script Language

زبان نسخه DPL از یک ساختار دستوری کاملاً شبیه به زبان برنامه نویسی C++ استفاده می‌کند. این نوع زبان، به راحتی قابل خواندن و یادگیری می‌باشد. این مجموعه فرمان اصلی (مبنا) در کوچکترین اندازه ممکن نگهداری شده و باقی مانده است.

ساختار مذکور به قسمت های زیر تقسیم شود :

- تعاریف متغیر
- تخصیصها و عبارات
- دستورات اجرای برنامه
- روش فراخوانی ها

این عبارات در یک نسخه DPL بوسیله نقطه ویرگولها (semicolons or ;) از هم جدا شده و همچنین بوسیله آکولادها با هم دسته بندی می‌شوند.

مثال :

```
statement1;
statement2;
if (condition) {
groupstatement1 ;
groupstatement2;
}
```

### ۱۵-۲-۱ تعاریف متغیر

DPL از انواع پارامترهای داخلی زیر استفاده می‌کند :

- **double** : عدد حقیقی ۱۵ رقمی
- **Int** : عدد صحیح
- **Object** : مرجعی برای عناصر DIgSILENT
- **Set** : آرایه ای از عناصر

ماتریسها و بردارها به عنوان عناصر خارجی موجود هستند.  
این ساختار دستور برای تعیین متغیرها به صورت زیر می باشد:

[VADREF]	=	[TYPE ] Var name.Varname ... ,Varname;
[TYPE ]	=	double   int   object   set

تعاریف و اعلام همه پارامترها باید با هم در اولین خطوط ابتدای این نسخه DPL داده شوند. این نقطه ویرگول اختیاری است.

چند مثال:

```
double Losses, Length, Pgen;
int NroFBreakers, i, j;
object 01, 02, BestSwitchToOpen;
set AllSwitches , AllBars;
```

### ۱۵-۲-۲ عبارات جایگزینی و عبارات

ساختار دستور زیر برای اختصاص یک اندازه برای یک متغیر استفاده می شود:

Variable	=	expression
Variable	-=	expression
Variable	+=	expression

عبارت جایگزینی جمع " += " مقدار سمت راستی را به این متغیر اضافه کرده و عبارت جایگزینی منفی " -= " مقدار سمت راستی را از آن کم می کند.

برای مثال:

```
double x,y;
x = 0.5*pi( ) ; ! x now equals 1.5708
y = sin(x) ; ! y now equals 1.0
x += y ; ! x now equals 2.5708
y - = x ; ! y now equals -1.5708
```

توابع و عملکردهای زیر قابل استفاده می باشند:

- عملکردهای حساب شامل: +, -, \*, /,
- توابع استاندارد: که عبارتند از:

sin (x)	cos (x)	tan(x)	asin(x)
acos(x)	atan(x)	sinh(x)	cosh(x)
tanh(x)	exp(x)	In(x)	log (x)(basis 10)
abs (x)	min (x,y)	max(x,y)	sqrt (x) (square root)
trunc(x)	frac(x)	round(x)	sqr(x) (power of 2)
pow (x,y)	modulo(x,y)	ceil (x)	floor(x)
rand ()			

توابع مثلثاتی بر اساس رادیان (RAD) قرار داده می‌شوند.  
تابع "rand()" به یک عدد اتفاقی توزیع شده یکنواخت در دامنه [0,1] بر می‌گردد.

### ۱۵-۲-۳ دستورالعمل های روند برنامه

فرامین روند (جریان) زیر موجود هستند :

if	( [boolexpr] )	[statlist ]		
if	( [boolexpr] )	[statlist ]	else	[statlist]
do	[statlist ]	while	( [boolexpr] )	
while	( [boolexpr] )	[statlist ]		

که در آن :

[boolexpr] = expression	[boolcomp]	expression
[boolcomp] = <   >   =   >=   <=   <>   !=		
[statlist ] = statement ;   { statement; [statlist] }		

- عملکردهای یکسانی: .not.
- عملکردهای باینری (دودویی): .and..or..nand..nor..eor.
- پرانتزها {عبارات منطقی}

برای مثال :

```
if (a<3)    b = a*2;
else       b = a / 2;
```

```
while (sin (a) >= b * c) {
  a = 0 : dline ;
  c = c + delta ;
}
if ({. not. a }. and. { b <>3}) {
  err = ldf. Execute ( ) ;
  if (err) {
    Ldf : iopt - lev = 1 ;
  }
}
```

```

err = Ldf. Execute ( ) ;
Ldf : iopt - lev = 0 ;
}
}

```

### Break and Continue

دستور حلقه ای "while-do" و "do-while" ممکن است دارای فرامین Break و Continue باشند. فرامین Break و Continue در بیرون یک دستور حلقه ای ظاهر نمی‌شوند.

فرمان Break به کوچکترین دستور پیوستی، دستور "while-do" یا "do-while" پایان می‌دهد. اجرای نسخه DPL با اولین فرمان متعاقب این دستور حلقه ای ادامه پیدا می‌کند.

فرمان Continue سریعاً از اجرای دستورات بعدی در مورد کوچکترین دستور پیوستی دستور "while-do" یا "do-while" گذشته و آنها را نادیده می‌گیرد. اجرای نسخه DPL با ارتقاء عبارت حساب دودویی (0)، (1) این دستور حلقه ای ادامه می‌یابد. فهرست این دستور حلقه ای در زمانی که این عبارت به TRUE ارتقاء پیدا می‌کند، دوباره اجرا می‌شود.

در غیر این صورت این دستور حلقه ای پایان داده شده و این اجرا با اولین فرمان که بعد از دستور حلقه ای می‌آید، ادامه پیدا خواهد کرد.

مثالها :

```

01 = S1. First ( ) ;
while (01) {
    01. Open ( ) ;
    err = Ldf. Execute ( ) ;
    if (err) {
        ! skip this one
        01 = S1. Next ;
        continue ;
    }
    02 = S2. First ( ) ;
    AllOk = 1 ;
    DOReport ( 0 ) ; !reset
    while ( 02 ) {
        err = Ldf. Execute ( ) ;
        if (err) {
            ! do not continue
            AllOk = 0 ;
            break;
        } else {
            DOReport ( 1 ) ; ! add
        }
        02 = S2. Next ( ) ;
    }
    if (AllOk) {
        DOReport (2) ; ! report
    }
    01 = S1. Next ( ) ;
}

```

## ۱۵-۲-۴ ورودی و خروجی

فرمان "input" از کاربر می‌خواهد که یک مقدار را وارد کند.

```
input(var , string);
```

فرمان "input" یک پنجره را با یک رشته و یک خط ورودی که در روی آن کاربر می‌تواند یک اندازه را وارد کند، باز می‌نماید. این مقدار می‌تواند به این متغیر "var" اختصاص داده شود. فرمان "Output" خطی از یک متن را در پنجره خروجی می‌نویسد.

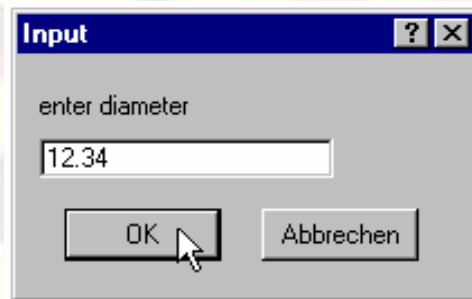
```
Output(String);
```

این رشته می‌تواند دارای علائم "=" که در پی آن نام متغیر قرار گرفته، باشد. پس از آن نام این متغیر با اندازه آن جایگزین می‌شود.

مثال:

```
input(diameter , 'enter diameter');
output('the entered Value = diameter');
```

نتایج این مثال در پنجره باز شده که در شکل ۱۵-۲ هم ترسیم شده، نشان داده می‌شود.



شکل ۱۵-۲: پنجره ورودی

متن بعدی در پنجره خروجی ظاهر می‌شود.

DIgSI/dpl - the entered value = 12/3400

### 15.3 Access to Other Objects

با کمک این ساختار دستور برای تعاریف پارامتر، روند برنامه و ورودی و خروجی اکنون امکان ایجاد یک برنامه کوچک وجود دارد. به هر حال چنین نسخه‌ای توانایی استفاده یا به کار بردن ماهرانه متغیرهای موضوعات خارجی را ندارد. به طور مثال امکان نوشتن یک نسخه که به منظور انتخاب بهترین نوع خط، یک نوع خط خاص را بوسیله جانشین‌های بهتر دیگر جایگزین می‌کند، وجود ندارد. چنین نسخه‌ای باید بتواند به موضوعات خاص (خطوط) و مجموعه‌های خاص موضوعات (مجموعه انواع خطوط دیگر) دست پیدا کند.

زبان DPL دارای چندین روش است که با آنها موضوعات پایگاه داده و پارامترهای آنها در نسخه DPL حاضر

می‌شوند:

- اولین روش مستقیم باید ایجاد موضوع یا مرجعی برای یک موضوع در خود پوشه فرمان باشد. چنین موضوعی به صورت مستقیم به عنوان متغیر "Object" در این نسخه موجود می‌باشد. نام این متغیر؛ نام این موضوع در پایگاه داده می‌باشد.
- انتخاب کلی باید استفاده شود. این روش فقط وقتی مفید است که دستوری که در آن به موضوعات دسترسی پیدا می‌کنند، مهم نباشد. وقتی یک انتخاب یا در گرافیک تک خطی یا مرورگر پایگاه داده کلیک راست شده و گزینه "Execute DPL Script" انتخاب می‌شود.
- این انتخاب کلی بطور خودکار پر می‌شود.
- اساساً فهرست موضوعات خارجی وقتی که یک نسخه باید برای موضوعات یا انتخابات خاص اجرا شود، استفاده می‌شوند. این فهرست موضوعات خارجی، بیش از یک فهرست نامهای مستعار "alias" نیستند. این موضوع خارجی جهت انتخاب موضوعات خاص برای هر نام مستعار قبل از اجرای این نسخه استفاده می‌شود.

### ۱۵-۳-۱ متغیرهای موضوع و روشها

اگر یک موضوع پایگاه داده با فرمان DPL شناخته شده باشد، بعد از آن می‌تواند همه روشهای آنرا فرا خوانده و همه متغیرهای آن در دسترس قرار گیرند. به طور مثال اگر بخواهید فرمان پخش بار را به منظور وارد کردن یک محاسبه پخش بار نامتقارن تغییر دهید، می‌توانید پارامتر "iopt\_net" را عوض نمایید. این روش با استفاده از یک عبارت جایگزینی انجام شده است:

```
Ldf : iopt_net = 1 ; ! force unbalanced
```

در این مثال موضوعات پخش بار به عنوان متغیر موضوعات "Ldf" شناخته می‌شود. ساختار دستور کلی برای یک پارامتر موضوع پایگاه داده به صورت زیر می‌باشد:

```
Object name : parameter name
```

با همین روش امکان دسترسی به یک اندازه از یک موضوع پایگاه داده وجود دارد. به طور مثال می‌توان به یک نتیجه از محاسبات پخش بار رسید. یکی از این نتایج، موضوع بارگیری از یک خط است که در متغیر "c:loading" ذخیره می‌شود. مثال زیر پخش بار نامتعادل را اجرا کرده و گزارشی از بارگیری خط را به ما می‌دهد:

```
00. int error ;
01. double loading ;
02. Ldf : iopt - net = 1 ; ! force unbalanced
03. error = Ldf. Execute ( ) ; ! execute Load flow
04. if (error) {
05.   exit ( ) ;
06. } else {
07.   loading = Line : c : loading ; ! get Line Loading
```

```
08. output ( 'loading = loading' ); ! report Line Loading
```

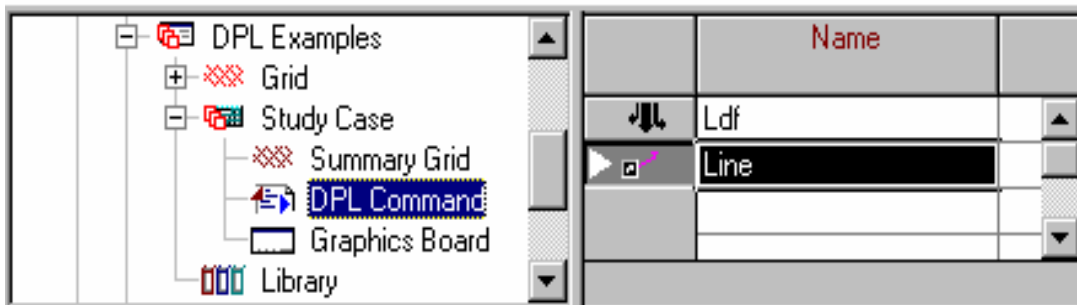
```
09. }
```

این مثالها خیلی ابتدایی و اولیه هستند، اما روشهای اساسی و اصولی را برای مفاهیم پایگاه داده ها و پارامترهای دیگر نشان می دهد.

## 15.4 Access to Locally Stored Objects

دستیابی به موضوعات ذخیره شده محلی فقط در صورتی امکان پذیر است که نام این موضوع همانند نام متغیر در نسخه تعریف شود. به طور مثال امکان دستیابی به موضوعی که نام آن "My Loadflow\~{}1\*" است، وجود ندارد.

با مثالی که در شکل ۱۵-۳ نشان داده شده، یک نسخه DPL در سمت چپ نشان داده می شود که دارای فرمان پخش بار بوده و یک مرجع برای خطی که در پوشه محتویات آن است در سمت راست دیده می شود.



شکل ۱۵-۳: محتویات DPL

هم اکنون این مثال نسخه DPL بطور مستقیم به این موضوعات همانند موضوعات "Ldf" و "Line" دستیابی پیدا می کند. در مثال صفحه بعد موضوع "Ldf" یک فرمان پخش بار است که در خط (01) برای اجرای پخش بار استفاده می شود:

```
00. int error
01. error = Ldf. Execute( ) ;
02. if (error) {
03.   output ( ' Load flow command returns an error ' ) ;
04.   exit( ) ;
05. }
```

در خط (01) پخش بار بوسیله فراخوانی روش "Execute( )" مربوط به فرمان پخش بار محاسبه می شود. جزئیات فرمان پخش بار، همانند انتخاب محاسبه پخش بار بین یک تکفاز متعادل یا یک سه فاز نامتعادل با ویرایش موضوع "Ldf" در پایگاه داده انجام می شود. بسیاری از مفاهیم دیگر در پایگاه داده دارای روشهایی هستند که می توان آنها را از نسخه DPL فراخواند.

محتویات DPL همچنین برای گنجاندن نسخه های آن در داخل نسخه های دیگر و بعد از آن ایجاد زیر روالهای DPL استفاده می شوند.

## 15.5 Accessing the General Selection

دستیابی موضوعات پایگاه داده با ذخیره کردن آنها یا مرجعی برای آنها در فرمان DPL اگر بخواهیم به بیشتر موضوعات مثلاً جستجوی یک خط با بارگیری بالا دستیابی پیدا کنیم باعث ایجاد مشکل می‌شود. ایجاد یک مرجع برای یک خط و خطوط بیشتر غیر ممکن است. یک روش دقیق تر، استفاده از انتخاب کلی DPL و پر کردن آن برای همه خطوط می‌باشد. مرورگر پایگاه داده چندین روش برای پر کردن انتخاب DPL با کمترین تلاش و کار را ارائه می‌کند. این انتخاب می‌تواند برای دستیابی غیر مستقیم به هر خط بوسیله یک متغیر "object" مربوط به DPL استفاده شود. در این روش می‌توان یک حلقه را ایجاد کرد که در آن بتوان بیشترین بارگیری را جستجو کرد. این مسئله در این مثال نشان داده می‌شود:

```

00.int error ;
01. double max;
02. object O , omax;
03. set S ;
04.
05. error = Ldf .Execute( ) ;           ! execute a loadflow
06. if ( error ) exit( ) ;             ! exit on error
07.
08. S = SEL .AllLines( ) ;             ! get all selected lines
09. Omax = S.First( ) ;                ! get first line
10. if ( Omax ) {
11.   max = Omax:c:loading ;           ! initialize maximum
12. } else {
13.   output ( 'No lines found in selection ' ) ;
14.   exit ( ) ;                       ! no lines : exit
15. }
16. O = S .Next ( ) ;                  ! get next line
17. while ( O ) {                       ! while more lines
18.   if ( O:c:loading>max ) {
19.     max = O:c:loading ;             ! update maximum
20.     omax = O ;                      ! update max loaded line
21.   }
22.   O = S .Next ( ) ;
23. }
24. output ('max loading=max for line') ; ! output results
25. Omax. ShowFullName ( ) ;

```

موضوع "SEL" استفاده در خط (08)، متغیر موضوع ذخیره شده است که با گزینه "General Selection" در کادر محاوره ای فرمان DPL برابری می‌کند. موضوع "SEL" در همه مواقع در همه نسخه های DPL موجود می‌باشد و فقط یک موضوع "General Selection" تک در یک زمان برای همه نسخه ها معتبر می‌باشد. این جمله بدین معنا می‌باشد که تنظیم "General Selection" در کادر محاوره ای فرمان DPL، این گزینه را برای همه فرمانهای DPL دیگر هم تغییر می‌دهد.

روش "All Line( )" در خط (08) به مجموعه‌ای از خطوط پیدا شده در این انتخاب کلی "General selection" بر می‌گردد. این مجموعه به متغیر "S" اختصاص داده می‌شود. هم اکنون می‌توان به

تک تک این خطوط با استفاده از روشهای مجموعه "First ( )" و "Next( )" در خطوط 09، 16، 22 دست یافت.

این خطوط با بالا ترین بارگیری در متغیر "Omax" نگهداری می شود. نام و موقعیت پایگاه داده با فراخوانی "Show Full Name ( )" در پنجره خروجی، انتهای این نسخه نوشته می شود.

## 15.6 Accessing External Objects

محتویات DPL امکان دستیابی به موضوعات خارجی در نسخه DPL را امکان پذیر می سازد. موضوع انتخاب کلی خاص "SEL" برای دادن همه توابع DPL و دستیابی توابع آنها در یک انتخاب مرکزی موضوعات استفاده می شود.

بنابراین وقتی باید به بیش از یک موضوع خاص در این نسخه دسترسی پیدا کرد، این روش انعطاف پذیر باعث ایجاد مشکلاتی می شود. با ایجاد مراجعی برای این موضوعات در خود فرمان DPL، این فرمان مختص حالت محاسبه جاری می شود. جمع آوری موضوعات در انتخاب کلی مشکل، موضوع جدیدی را به وجود می آورد. برای جلوگیری از ایجاد فرمانهای محاسبه خاص توصیه می شود که محتویات DPL را برای همه موضوعاتی که در واقع به نسخه DPL و آنهايي که مستقل از مکان و چگونگی استفاده از این نسخه می باشند، را ذخیره نمایید. مثالهای مناسب این موضوع فرامین پخش بار و اتصال کوتاه یا موضوعات بردار یا ماتریس که فرمان DPL برای محاسبه آنها استفاده می شود، می باشند.

اگر نسخه DPL باید به موضوع پایگاه داده مستقل از مکان و چگونگی استفاده از این نسخه دست پیدا کند، یک "External object" (موضوع خارجی) باید به فهرست موضوع خارجی در فرمان ریشه DPL اضافه شود. چنین موضوع خارجی یک مرجع نامگذاری شده برای یک موضوع پایگاه داده خارجی می باشد. این موضوع بوسیله آن نام ارجاع داده می شود. از این رو تغییر این موضوع مسئله انتخاب موضوع دیگر می باشد. در شکل ۱۵-۴ مثالی از یک موضوع خارجی داده می شود. این موضوع خارجی را می توان در این نسخه DPL با نام "Bar1" همانطور که در این مثال نشان داده شده، ارجاع داد.

مثال :

sagdepth = Bar1:u ;

	Name	object	Description
▶ 1	Bar1	B1	Busbar for which the sag table is assessed

شکل ۱۵-۴ : جدول موضوع خارجی DPL

## 15.7 Remote Scripts and DPL command Libraries

ساده ترین روش برای توسعه فرمان DPL جدید باید ایجاد یک *Com DPL* جدید در حالت مطالعه فعال جدید و نوشتن مستقیم این نسخه در آن موضوع DPL می باشد. در چنین روشی یک "root command" (فرمان ریشه) DPL ایجاد می شود. اگر فرمان ریشه به زیر روالهای DPL نیاز داشته باشد، می توان یک یا چند موضوع فرمان DPL را در محتویات آن ایجاد کرد. هر یک از این زیر روالها به صورت عادی همانند، توابع ریشه هم نوشته می شوند.

فرمان جدید DPL نوشته شده با زیر روال آنرا می توان بررسی کرده و در مورد مطالعاتی "study case" فعال جاری استفاده کرد. به هر حال زمانی که حالت مطالعه دیگری فعال است، این فرمان جدید را نمی توان اجرا کرد. به منظور استفاده فرمان DPL در حالت های مطالعه دیگر یا حتی در پروژه های دیگر باید فرمان DPL و محتویات آن کپی شوند. این کار باعث می شود که امکان تغییر و تعویض فرمان DPL بدون تغییر دادن حتمی همه کپی های آن غیر ممکن شود.

راه حل مشکل استفاده از "remote scripts" (اسکرپت های راه دور) می باشد. روال ایجاد استفاده از اسکرپت های راه دور بعداً توضیح داده می شود.

فرض کنید که یک فرمان DPL جدید را در حالت مطالعه فعال جاری ایجاد کرده و چک کرده ایم. ما می خواهیم این فرمان DPL را در یک مکان ذخیره سازی ذخیره کرده و امکان استفاده از آن را در حالت های مطالعه و پروژه های دیگر ایجاد نمائیم.

این عمل با مراحل زیر انجام می شود:

- فرمان DPL را در پوشه کتابخانه کپی کنید. این عمل باعث کپی شدن محتویات فرمان DPL یعنی با همه زیر روالهای آن و موضوعات ذخیره شده محلی دیگر می شود.
- فرمان DPL کپی شده را بوسیله تنظیم مجدد همه موضوعات خارجی خاص پروژه تعمیم دهید. همه اندازه های متغیر رابط را در اندازه های پیش فرض آنها تنظیم کنید. برای جلوگیری از حذف قسمتی از فرمان DPL مطمئن شوید که اگر هر فرمان یا زیر روال DPL به یک اسکرپت راه دور بر می گردد، همه آن اسکرپت های راه دور هم در این پوشه کتابخانه ذخیره شوند.
- حالت مطالعه دیگری را فعال کنید.
- موضوع فرمان DPL جدید (ComDPL) را در حالت مطالعه فعال ایجاد کنید.
- مرجع "DPL script" را برای فرمان DPL کپی شده تنظیم کنید.
- موضوعات خارجی لازم را انتخاب نمائید.
- به صورت دلخواه اندازه های پیش فرض متغیرهای رابط را تغییر دهید.
- دکمه **Check** را برای بررسی اسکرپت DPL، فشار دهید.

دکمه **Check** یا **Execute** همه قسمت های اسکرپت راه دور را که برای اجرا لازم می باشد، در این کتابخانه کپی می کنند. این قسمت ها شامل زیر روالهایی که به اسکرپت های راه دور ارجاع می شوند، موضوعات فرامین و همه موضوعات می باشند. بعضی موضوعات طبقه ها به عنوان مرجع کپی شده، دیگر طبقات بطور کامل کپی می شوند.

فرمان DPL جدید دارای یک اسکریپت نمی‌باشد اما این اسکریپت راه دور را اجرا می‌کند. برای اجرای خود آن، هیچ تغییری ایجاد نمی‌شود. به هر حال بیش از یک فرمان DPL را می‌توان به همان اسکریپت راه دور ارجاع داد. تغییر اسکریپت راه دور یا هر قسمت موضوعات محلی آن یا زیر فرمانها اجرای همه فرامین DPL را که به آن ارجاع می‌شوند، تغییر می‌دهد.

### ۱۵-۷-۱ زیر روالها و فراخوانی قراردادها

موضوع فرمان DPL می‌تواند دارای محتویات فرمان DPL دیگر باشد. در این حالت، این زیر روال "subroutine" DPL به حساب آمده را می‌توان در اسکریپت فرمان DPL پیوستی فراخواند. در اصل این فراخوانی فرقی با فراخوانی مثلاً یک فرمان پخش بار از یک اسکریپت DPL ندارد. بر عکس بیشتر موضوعات فرامین دیگر، فرمان DPL فقط دارای یک روش می‌باشد:

**( Execut int ;** نسخه DPL را اجرا می‌کند.

تفاوت در این است که هر زیر روال DPL دارای پارامترهای رابط متفاوتی است که می‌توان آنها را بوسیله فراخوانی تغییر داد. پارامترهای رابط را می‌توان بوسیله تهیه یک یا چند شناسه فراخوانی بطور مستقیم در همین زمان فراخوانی هم تنظیم کرد. شناسه های فراخوانی بر طبق ترتیب ظهور به پارامترهای رابط اختصاصی داده می‌شوند. مثال زیر این مسئله را روشن می‌کند.

فرض کنید که یک زیر فرمان DPL، "Sub1" با بخش رابط همانطور که در شکل ۱۵-۵ ترسیم شده، دارید.

Interface parameter:					
	Type	Name	Value	Unit	Description
1	double	step	2.5	MW	Step transfer
2	object	Line			Transfer line
▶ 3	set	Outages			Objects for (n-1) check

شکل ۱۵-۵: بخش رابط زیر روال

بنابراین فرمان فراخوانی را می‌توان استفاده کرد، برای مثال:

```
! set the parameters:
Sub1 : step    = 5.0;
Sub1 : Line    = MyLine;
Sub1 : Outages = MySelection;
! execute the subroutine:
error = Sub1.Execute ( );
```

به هر حال با استفاده از شناسه های فراخوانی می‌توانیم به صورت زیر بنویسیم:

```
! execute the subroutine:
MySelection );          , MyLine ,error = Sub1. Execute (5.0
```

## 15.8 DPL Functions and Subroutines

ساختار دستور DPL بسیار مختصر می‌باشد، زیرا اساساً هدف آن کار کردن عملکردهای اصلی مانند محاسبات ساده، انتخابهای if-then-else، حلقه های do-while و غیره می‌باشد.

قدرت و استحکام زبان DPL امکان فراخواندن توابع و ایجاد زیر روالها می‌باشد. یک تابع را که می‌توان بوسیله فرمان DPL فراخواند، یک روش نامیده می‌شود.

چهار نوع روش تشخیص داده می‌شوند که عبارتند از :

**Internal methods** : اینها روشهای ایجاد شده فرمان DPL هستند. آنها را می‌توان همیشه فراخواند.

**Set methods** : این روشها برای متغیرهای "set" مربوط به DPL موجود هستند.

**Object methods** : این روشها برای متغیرهای (object) مربوط به DPL موجود می‌باشند.

**External methods** : این روشها برای موضوعات *DIgSILENT* خارجی معینی از قبیل فرمان پخش بار، موضوع خط، ماشین همزمان و غیره موجود هستند.

همه این روشها در بخشهای زیر شرح داده شده است.

### ۱۵-۸-۱۵ روشهای داخلی DPL

زبان برنامه DPL دارای یک مجموعه کوچک از فرامین داخلی خاص DPL می‌باشد :

Random	یک شماره اتفاقی را بر می‌گرداند
Valid DF	نتیجه یک پخش بار معتبر را کنترل می‌کند
Valid RMS	نتیجه یک همانند سازی معتبر را کنترل می‌کند
ValidSHC	نتیجه یک اتصال کوتاه را بررسی می‌کند
Valid SIM	نتیجه یک همانند سازی معتبر را کنترل می‌کند
All Relevant	همه موضوعات مرتبط محاسبه را بر می‌گرداند
ActiveCase	حالت محاسبه فعال را بر می‌گرداند
SumMaryGrid	شبکه مختصر را بر می‌گرداند
Active Project	این پروژه فعال را باز می‌گرداند
Write	یک گزارش می‌نویسد
ResetCalculation	محاسبات را دوباره تنظیم می‌کند
PostCommand	یک فرمان را به مسیر این فرمان اضافه می‌کند
Exe	یک فرمان را اجرا می‌کند
ClearCommands	مسیر فرمان را پاک می‌کند
GetLanguage	زبان جاری را بر می‌گرداند
Delete	موضوع را حذف می‌کند

برای اطلاعات بیشتر درباره این دستورات می‌توانید به راهنمای **on-line** مراجعه کنید.

### ۱۵-۸-۲ روشهای مجموعه DPL

روشهای مجموعه دارای توابعی برای پارامترهای نوع مجموعه هستند :

**set. [SETMETHOD] ( arguments ) ;**

روشهای [SETMETHOD] در زیر موجود هستند :

Add	اضافه کردن عنصر
Remove	حذف کردن عنصر
Count	برگشت به عناصر ذخیره شده
First	برگشت به عناصر اولیه
Next	برگشت به عنصر بعدی
Firstmatch	برگشت دادن اولین مورد تطبیق داده شده
Netmatch	برگشت دادن مورد بعدی تطبیق داده شده
SortToVar	مرتب کردن عناصر براساس نوع متغیر
SortToClass	مرتب کردن عناصر براساس کلاس هایشان
SortToName	مرتب کردن عناصر براساس نام هایشان
MarkInGraphics	علامت گذاری در نمودارها

برای اطلاعات بیشتر درباره این دستورات می‌توانید به راهنمای **on-line** مراجعه کنید.

### ۱۵-۸-۳ روشهای موضوعی

روشهای موضوعی برای هر نوع از طبقه موضوعات خاص هستند. به طور مثال یک موضوع فایل نتیجه **(ElmRes)** دارای یک روش **"Write"** که هیچ مفهومی برای یک موضوع فرمان پخش بار ندارند. ساختار دستور کلی برای یک روش موضوعی همانند ساختار دستور روش مجموعه می‌باشد :

**object. [OBJMETHOD] ( arguments ) ;**

ذیلاً کلیه روشهای عمومی [OBJMETHOD] قابل دسترس برای تمامی طبقه ها بصورت کلی لیست شده

است :

Unom	ولتاژ نامی را بر می‌گرداند
Markin Graphics	موضوع را در این گرافیک علامت گذاری می‌کند
Show Full Name	مسیر کامل پایگاه داده و نام را چاپ می‌کند
IsClass	یک طبقه معین را کنترل می‌کند
Add Copy	یک کپی از یک موضوع را اضافه می‌کند

Create Object

موضوع جدیدی را ایجاد می کند

Edit

کادر محاوره ای این موضوع را باز می کند

برای اطلاعات بیشتر درباره این دستورات می توانید به راهنمای on-line مراجعه کنید.

کلیه روشهای خاص [OBJMETHOD] موجود بصورت کلی لیست شده است. فراخوانی این روشها برای طبقه های نامناسب (به شکل غلط) منجر به صدور خطا خواهد شد :

ComRes.ExportFullRange	ComTopo.IsRadial
ComEcho.Execute	ComEcho.On
ComEcho.Off	ComTime.Execute
Set Filt.Get	ComDPL.Execute
Int Mat.Get	IntMat.Set
IntMat.Init	IntMat.Resize
IntMat.Nrow	IntMat.NCol
IntMat.RowLbl	IntMat.ColLbl
IntVec.Get	IntVec.Set
IntVec.Init	IntVec.Resize
IntVec.Size	ElmCoup.Close
ElmCoup.Open	ElmCoup.IsOpen
ElmCoup.IsClosed	ElmLne.HasRoutes Or Sec
ElmLne.Get Type	ElmLne.IsCable
ElmLne.IsNetCoupling	ElmLne.SetCorr
TypLne.IsCable	ElmLne.SetNomCurr
ElmRes.Clear	ElmRes. Write
ElmRes.Draw	ElmRes. WriteDraw
ComRel3.Execute	ComInc.Execute
Comldf.Execute	ComShc.Execute
StaSwitch.Close	StaSwitch.Open
StaSwitch.IsOpen	StaSwitch.IsClosed
SetFeeder.GetAll	SetFeeder.GetBuses
SetFeeder.GetBranches	SetPath.GetAll
SetPath.Get Busses	SetPath.GetBranches
SetPath.AllBreakers	SetPath.AllClosedBreakers
SetPath.AllOpenBreakers	SetSelect.AddRef
SetSelect.Clear	Set Selecte.AllElm

SetSelect.AllLines	SetSelect.AllBars
SetSelect.AllLoads	SetSelecte.AllAsm
SetSelect.AllSym	SetSelecte.AllTyplne
Set Select.All	Set Select.GetAll
SetSelect.Allbreakers	SetSelecte.AllClosedBreakers
Set Select.AllOpenBreakers	IntForm.SetText

برای اطلاعات بیشتر درباره این دستورات می توانید به راهنمای [on-line](#) مراجعه کنید.



شرکت برق منطقه ای فارس