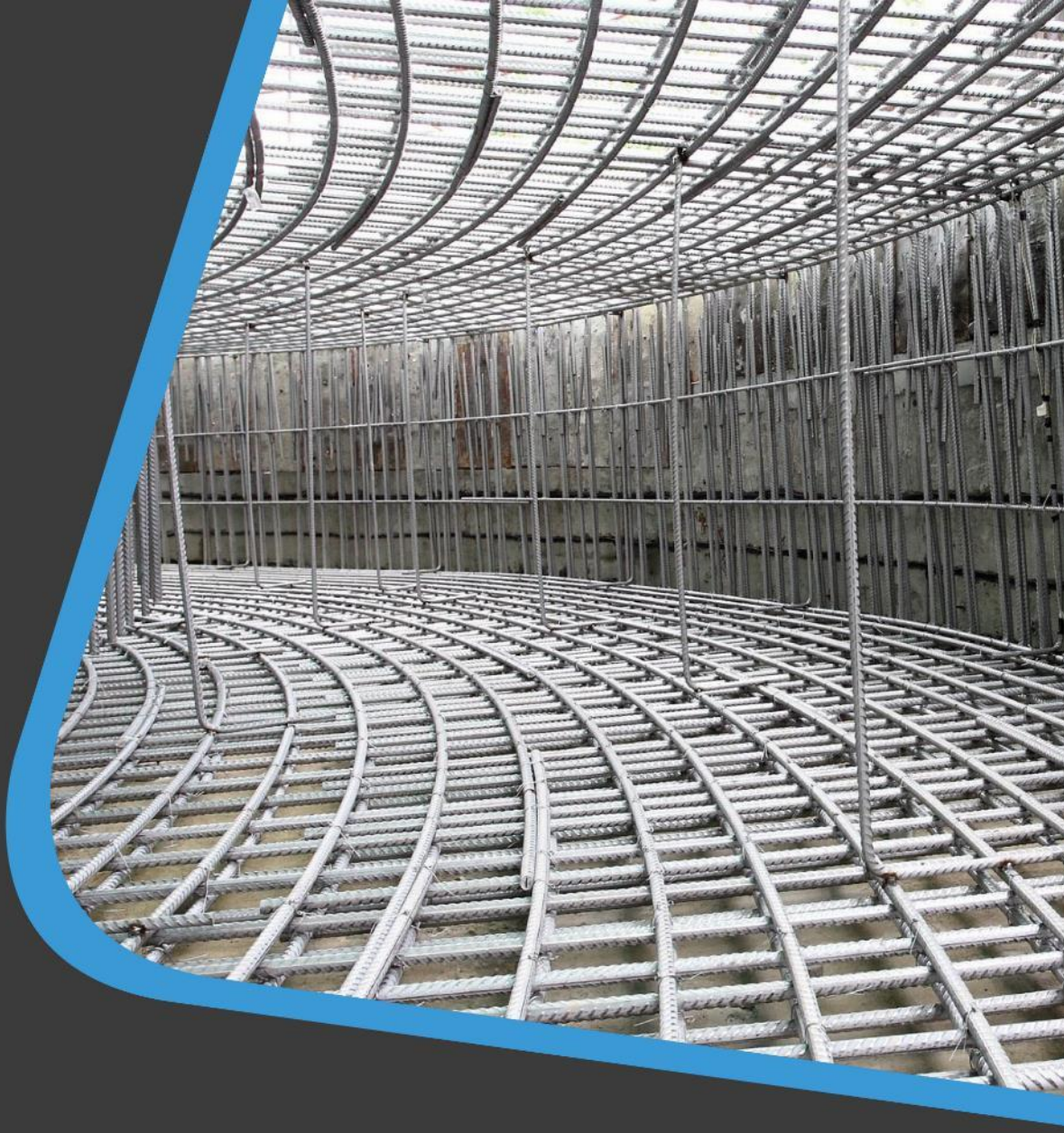


RebarCAD
LISP Programming Interface Guide

CADS
GLOBAL CONSTRUCTION
SOFTWARE AND SERVICES

AUTODESK
Authorised Developer

Microsoft
Partner

Revision history

Date	Version	Description
Feb 2006	1.0	Compatible to RebarCAD/CADS RC 8.4
July 2019	2.0	Compatible to RebarCAD/CADS RC 2020.0
July 2020	3.0	Compatible to RebarCAD/CADS RC 2021.0
Dec 2020	4.0	Compatible to RebarCAD/CADS RC 2021.10
June 2021	5.0	Compatible to RebarCAD/CADS RC 2022.0
Jan 2022	6.0	Compatible to RebarCAD/CADS RC 2022.10
Jan 2023	7.0	Compatible to RebarCAD/CADS RC 2023.10
Jun 2023	8.0	Compatible to RebarCAD/CADS RC 2024.0
Sep 2023	8.1	Compatible to RebarCAD India 2024.0
Nov 2024	9.0	Compatible to RebarCAD India 2025.0
Dec 2025	10.0	Compatible to RebarCAD India 2026.0

RebarCAD

Computer and Design Services Limited
Arrowsmith Court,
10, Station Approach
Broadstone
Dorset, BH18 8AX

Important Notice
Please read this –

All due care has been taken to ensure that the data produced by this program is accurate. However, it remains the responsibility of the user to verify that any design based upon this data meets all applicable standards. Please refer to the CADS Software Maintenance and License Agreement for detailed information.

Computer and Design Services Ltd.,
Arrowsmith Court,
Broadstone,
Dorset,
UK
BH18 8AX

Tel. (Sales): +44 (0) 1202 603031
Tel. (Support): +44 (0) 1202 603733
Fax: (Support): +44 (0) 1202 690284
Email (Sales): sales@cads.co.uk
Email (Support): support@cads.co.uk
Website: http://www.cadsglobal.com

Contents

1 Introduction

1.1 Overview

RebarCAD makes reinforcement detailing and scheduling within the AutoCAD environment accurate and cost effective. It provides you with various customizable options so that you can fit the reinforcement detailing environment to suit your project. RebarCAD also provides you with Application programming interface (API's) so that you can program your reinforcement detailing library for repetitive usage and thereby increasing productivity.

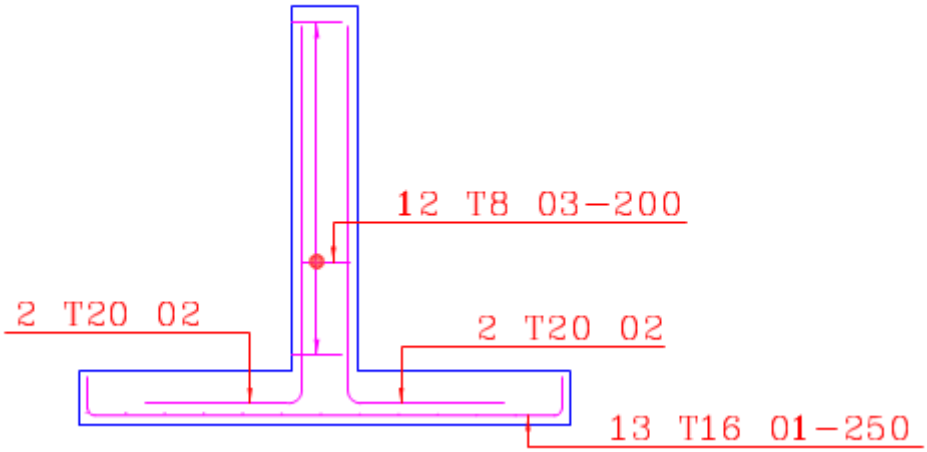
This is possible as RebarCAD exposes many of the important routines which can be used with AutoLISP scripting. With a basic knowledge of AutoLISP and RebarCAD routines you can create customizable routines for commonly used structural elements.

We will first take up a typical reinforcement detailing problem and explain to you its effectiveness. Later in the document you will learn that with the available RebarCAD API's you can have innumerable permutations and combinations to program almost all types of problems.

2 Getting Started

We commonly come across isolated footing detailing problems. The reinforcement placement is generally typical with few alterations based on the project and the designer's need.

For example, typical isolated footing will be detailed something like:



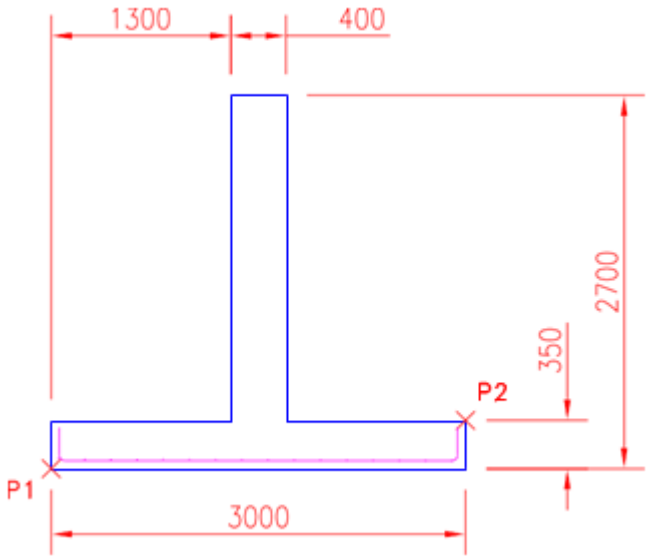
For the above mentioned footing diagram you need to draw the following:

- Bottom Bar
- Dowel Bars
- Range bar
- Bar Runs
- Link Bars

Generally, you need to draw a number of similar footings for Structural detailing. Instead of detailing similar reinforcements each time, you can create a small LISP routine using AutoLisp with the RebarCAD functions.

The complete routine to detail the above footing is available at the end of this document. For your better understanding, part of the routine used for detailing the bottom bar and the bar run is explained below.

In this program the Type, Grade, Shape code, Bar Handing, Bar Style are hard coded, i.e. we are using a set value. You can create a separate text file with all these details and it can be used for the program. The Sample Footing program uses a text file called Reinforcement.txt, which has all the bar details. By simply modifying the text file you can create different sets of bars.



2.1 Procedure to draw the reinforcement:

1. First draw the outlines with the required dimension.
2. Create a lisp file called demo.lsp in C:\
3. Copy and paste the code mentioned below to the demo.lsp
4. Save it and close.
5. In AutoCAD use the Appload command to load the demo.lsp
6. Type Demo in the command prompt. It will ask you to pick the Lower Left corner of the base and the upper right corner of the base. Pick points p1 & p2 as mentioned in the drawing.
7. Specify the Cover and Centre to centre spacing.
8. You can see the reinforcement drawn on the screen.
9. Repeat step 6 & 7 for different footing outlines.

2.2 Sample coding to draw the bottom bar and the bottom bar run.

```
;; function to draw the reinforcement details for the specified outlines
(defun c:demo ()

    ;; Std function call for each start and end of the program

    ;; This is a mandatory function call
    (CADS_RCL_ACAD_TOGGLE)

    ;; function call to display and set the member title
    (CADS_RCL_SET_MEMTITLE)

    ;; storing the old osnap mode so that it can revert back once the program is completed
    (setq old_osmode (getvar "osmode"))
```

```
;; to set the osnap mode end & intersection
(command "osmode" 33)

;; getting the corner points from the user

(setq    p1 (getpoint "\nPick lower left corner of the base: ")
         p2 (getpoint"\nPick upper right corner of the base: ")

)

;; restoring the old osnap mode
(command "osmode" old_osmode)

;; to set the dimension from the user input points

(setq Foun_Width (distance p1 (list (car p2) (cadr p1)))
     Foun_Depth (distance p1 (list (car p1) (cadr p2)))

)

;; to get the Cover & Centre to centre spacing value
(setq cover (getdist "\nEnter the Cover Value: ")

     ccspacing (getreal "\nEnter the Center to centre spacing :")

)

;; to set the insertion point of the bottom bar & setting the Leg dimensions of the bar
(setq inspt (list (+ (car p1) cover) (+ (cadr p1) cover))

       dimA (distance inspt (polar inspt (dtr 90) (- Foun_Depth cover cover)))
       dimB  (- Foun_Width cover cover)

       dimC dimA

)

;; converting all the Leg dimensions as a single string

(setq dimstring (strcat (rtos dimA) "," (rtos dimB) "," (rtos dimC)))

;;    Function    to    set    the    diameter    and    type
(CADS_RCL_SETTYPEDIAM "T" "16")

;; function call to draw the bar

(drawbar dimstring "21" inspt 0.0 "R" "Side")

;; to draw bottom bar runs

;; function to set bar runs as the view of the existing bar.
(CADS_RCL_NEWview Barename)

;; to set the range start point

(setq Rangeinspt (polar inspt (dtr 90) (* 16 1.0)))

;; function to draw the bar run

(CADS_RCL_BAR_RUN "21" Rangeinspt (polar inspt (dtr 0) (- dimB cover)) ccspacing 0)

;; Std function call for each start and ending of the program

;; This is a mandatory function call
(CADS_RCL_ACAD_TOGGLE)

)
```

```
;; function to draw the bar
(defun drawbar(dimstring Shapecode inspt Barangle Handing Style)
        ;; function to set the Leg dimension before drawing the bar
        (CADS_RCL_FillDims dimstring)

        ;; function to draw the bar
        (CADS_RCL_DRAW_BAR_ND shapecode inspt Barangle Handing Style)
        ;; to get the entity name of the last drawn object
        (setq Barename  (entlast))
        ;; function set the drawn bar as new set. If this is not set, then the drawn bar will not work
        ;; as rc bar
        (CADS_RCL_NEWSET Barename)
)
;; function to convert degrees to radians
(defun dtr(angle)
  (setq rad (* (/ pi 180.0) angle))
)
```

The List of RebarCAD Exposed functions are mentioned in the next chapter which will guide you to create the customized lisp programs. All the examples mentioned in the document for the RebarCAD exposed functions are created with BS 8666.bdf.

3 Exposed Functions

3.1 CADS_RCL_ACAD_TOGGLE

Standard function call at the start and end of all functions.

3.1.1 Syntax:

(CADS_RCL_ACAD_TOGGLE)

3.1.2 Sample Code:

(CADS_RCL_ACAD_TOGGLE)

3.2 CADS_RCL_Audit

This function will invoke the drawing audit, Match bars, and Check RC database commands based on the Command Number.

3.2.1 Syntax:

(CADS_RCL_ Audit Command Number as integer)

Command Number value ranges from 1 - 3

If the value is 1 then it will invoke the drawing audit command

If the value is 2 then it will invoke the drawing match bar command

If the value is 3 then it will invoke the drawing Check RC Database command

3.2.2 Sample Code:

(CADS_RCL_ Audit 1)

3.3 CADS_RCL_ACC_REFRESH

This function will update the accessories list on the drawing.

3.3.1 Syntax:

(CADS_RCL_ACC_REFRESH)

3.3.2 Sample Code:

```
(CADS_RCL_ACC_REFRESH)
```

3.3.3 Sample Program:

```
(defun C:RefreshAccessories ()
  (CADS_RCL_ACAD_TOGGLE)
  (CADS_RCL_ACC_REFRESH)
  (CADS_RCL_ACAD_TOGGLE)
)
```

3.4 CADS_RCL_BAR_ENDS

This function invokes the End points of the bar. The end points will be stored in the variables called lisplink1 & lisplink2.

3.4.1 Syntax:

(CADS_RCL_BAR_ENDS Barename as ExistingBarEntityName)

3.4.2 Sample Code:

```
(setq Barename (cdr (car (entget (car (entsel))))))
(CADS_RCL_BAR_ENDS Barename )
Command: !lisplink1
(19648.6 10457.6)
Command: !lisplink2
(23302.4 10114.6)
```

3.5 CADS_RCL_BAR_RUN

This is the function to draw the bar run.

3.5.1 Syntax:

(CADS_RCL_BAR_RUN Shapecode as string, startpoint as point, endpoint as point, centre to centre distance as real, quantity as integer)

3.5.2 Sample Code:

(CADS_RCL_BAR_RUN "20" RunInsertionPoint (polar RunInsertionPoint (dtr 0) 2000) 100.0 0)

Limitation:

The quantity will not be considered. It is mandatory to specify the non zero value for the centre to centre distance

3.5.3 Sample Program:

```
(defun C:DrawBarwithRun ()
    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_SETTYPEDIAM "T" "20")
    (setq BarInsertionPoint (getpoint "\nPick Bar Insertion point : "))
    (CADS_RCL_DRAW_BAR "20" BarInsertionPoint 0.0 "L" "Side" 2000) (setq
    BarEname (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET BarEname)
    (CADS_RCL_NEWview Barename)
    (setq RunInsertionPoint (polar BarInsertionPoint (dtr 90) 300))
    (CADS_RCL_BAR_RUN "20" RunInsertionPoint (polar RunInsertionPoint (dtr 0) 2000)
    100.0 0) (CADS_RCL_ACAD_TOGGLE)
)
```

3.6 CADS_RCL_ BarRef

This function is to draw the bar Reference.

3.6.1 Syntax:

(CADS_RCL_BarRef BarEname as EntityNname, InsertionPoint as point)

3.6.2 Sample Code:

(CADS_RCL_BarRef Barename point1)

3.6.3 Sample Program:

```
(defun C:DrawBarwithRef ()
  (CADS_RCL_ACAD_TOGGLE)
  (CADS_RCL_SET_MEMTITLE)
  (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
  (CADS_RCL_SETTYPEDIAM "T" "20")
  (setq BarInsertionPoint (getpoint "\nPick Bar Insertion point : "))
  (CADS_RCL_DRAW_BAR "20" BarInsertionPoint 0.0 "L" "Side" 2000) (setq
  BarEname (cdr (car (entget (entlast)))))
  (CADS_RCL_NEWSET BarEname)
  (CADS_RCL_NEWview Barename)
  (setq BarRefPoint (getpoint "\nPick Bar Ref point : "))
  (CADS_RCL_BarRef Barename BarRefPoint)
  (CADS_RCL_ACAD_TOGGLE)
)
```

3.7 CADS_RCL_ CalcCcNum

This is the function to calculate the number of bars based on the Run length and centre to centre spacing. It will store the values in the variables Lisplink1 & Lisplink2.

3.7.1 Syntax:

(CADS_RCL_ CalcCcNum RunLength as real, NumberOfBars as integer, CenterToCenterDistance as real, MethodOfCalculation as integer)

3.7.2 Sample Code:

(CADS_RCL_CalcCcNum 1000 0 200 1)

Command: !lisplink1 6

Command: !lisplink2 200.0

Limitation:

Method calculation should be 1.

3.8 CADS_RCL_ Draw_Bar

This is the function to draw a bar.

3.8.1 Syntax:

(CADS_RCL_DRAW_BAR shape code as string, Insertion point as point, Rotation as real, handing as string, view as string, Dimension as real, Multiplier as integer, Spacing as real, Quantity as integer, Notes as string)

3.8.2 Sample Code:

(CADS_RCL_DRAW_BAR “20” ‘(0 0) 0.0 "R" "Plan" 2000)

(CADS_RCL_DRAW_BAR “37” ‘(0 0) 0.0 "R" "Plan" 2000 3000)

3.8.3 Sample Program:

```
(defun C:DrawBar ()
    (CADS_RCL_ACAD_TOGGLE)
    (setq BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
          Shapecode (getstring "\nEnter the shape Code :") Dimension
          (getreal "\nEnter the dimension: ")
    ) (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETTYPEDIAM "R" "16")
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No") (CADS_RCL_DRAW_BAR shapecode
    BarInsertionPoint 0.0 "R" "Plan" dimension) (setq Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.8.4 Sample Program:

```
(defun C:DrawBar ()
        (CADS_RCL_ACAD_TOGGLE)
        (setq
                BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
                Shapecode (getstring "\nEnter the shape Code :")
                BarType (getstring "\nEnter the bar type :")
                BarSize (getstring "\nEnter the bar size :")
        )
        (CADS_RCL_SET_MEMTITLE)
        (CADS_RCL_SETTYPEDIAM BarType BarSize)
        (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
        (CADS_RCL_DRAW_BAR shapecode BarInsertionPoint 0.0 "R" "Side" 0 100 0 0 0 0 0 0 0 0 8 0.0
5 "Test Notes")
        (setq Barename (cdr (car (entget (entlast)))))
        (CADS_RCL_NEWSET Barename)
        (CADS_RCL_ACAD_TOGGLE)
)
```

3.9 CADS_RCL_CHANGE_DET_STD

This is the function to change the current detailing standard set in a drawing to a different one.

3.9.1 Syntax:

(CADS_RCL_CHANGE_DET_STD DEFFilename as string)

3.9.2 Sample Code:

(CADS_RCL_CHANGE_DET_STD “EC2”)

(CADS_RCL_CHANGE_DET_STD “EC2.def”)

(CADS_RCL_CHANGE_DET_STD “C:\\Program Files\\CADS\\AutoCAD 2026\\CADS RC India 2026.0\\CADS-RC\\params\\EC2.def”)

3.9.3 Sample Program:

```
(defun C:ChangeDetailingStandard ()
```

```
(
(CADS_RCL_CHANGE_DET_STD
    "C:\\Program Files\\CADS\\AutoCAD
    2026\\CADS RC India 2026.0\\CADS-
    RC\\params\\EC2.def")
)
```

3.10 CADS_RCL_ Draw_Bar_ND

This function is also used to draw bars. In this function there is no need to pass the leg dimensions function. Leg dimensions should be passed using another function called CADS_RCL_FillDims. Before using CADS_RCL_ Draw_Bar_ND the CADS_RCL_FillDims should be used.

3.10.1 Syntax:

(CADS_RCL_DRAW_BAR_ND shapecode as string, Insertion point as point, Rotation angle as real, handing as string, view as string)

3.10.2 Sample Code:

```
(CADS_RCL_DRAW_BAR_ND "37" point1 0.0 "L" "Side")
```

3.10.3 Sample Program:

```
(defun C:DrawBar_ND ()
    (CADS_RCL_ACAD_TOGGLE)
    (setq BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
          Shapecode (getstring "\nEnter the shape Code :")
    )

    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETTYPEDIAM "R" "16")
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_FillDims "2000")
    (CADS_RCL_DRAW_BAR_ND Shapecode BarInsertionPoint 0.0 "L" "Side") (setq
     Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.11 CADS_RCL_ ENTGET

This function helps to get details of an RC object.

3.11.1 Syntax:

(CADS_RCL_ENTGET BarEname as EntityName, KeyWord as string)

The Keyword can be any one of the following string. Based on the keyword the value will be stored in the variable called lisplink1.

Sl. No	Keyword	Description
1	number_bars	Number of bars
2	number_of_ranges	Number of ranges
3	no_sets	Number of bar sets
4	ref_pnt	Insertion point
5	dir_vect	Direction vector
6	handing	Handing
7	shape_code	Bend type
8	dimA	dimA - A leg dimension
	dimB	dimB - B leg dimension
	dimC	dimC - C leg dimension
	dimD	dimD - D leg dimension
	dimE	dimE - E leg dimension
	dimF	dimF - F leg dimension
	dimG	dimG - G leg dimension
	dimH	dimH - H leg dimension
	dimJ	dimJ - J leg dimension
	dimR	dimR - R leg dimension
9	diam	Bar diameter
10	size	Bar size

11	ndiam	Nominal bar diameter
12	type	Bar grade
13	view	Bar view
14	profile	Bar profile
15	bmark	Bar mark
16	cc_space	Bar spacing
17	etype	Entity type
18	type_of_xdata	XData type
19	label_notes	Label notes
20	type_of_range	Range type
21	run_length	Run length
22	int_dist	Intermediate range length
23	range_option	Range option
24	calc_method	Calculation method
25	range_slope	Range slope
26	length1	First bar length
27	length2	Last bar length
28	total_bar_length	Total bar length
29	member_title	Member title
30	release_number	Release number
31	drawing_sheet_number	Drawing sheet number
32	hidden_multiplier	Hidden bar multiplier
33	member_qty	Member quantity
34	release_description	Release description
35	bar_set_number	Bar set number
36	length	Bar length

37	first_bar_length	The length of the first bar in a bar range
38	last_bar_length	The length of the last bar in a bar range
39	release_order_status	Release ordered status(Ordered/Un-ordeed)
40	sub_release	Sub release number for the Release
41	bid_item	Bid item
42	coupler_type1	Type of the coupler attached to the end1
43	coupler_manufacturer1	Manufacturer of the coupler attached to the end1
44	coupler_description1	Description of the coupler attached to the end1
45	coupler_type2	Type of the coupler attached to the end2
46	coupler_manufacturer2	Manufacturer of the coupler attached to the end2
47	coupler_description2	Description of the coupler attached to the end2
48	end_treatment1	Type of the end treatment attached to the couple end1 (Coupler/Thread)
49	end_treatment2	Type of the end treatment attached to the couple end2 (Coupler/Thread)
50	coupler_bid_item1	Bid item associated with the coupler attached to the end1
51	coupler_bid_item2	Bid item associated with the coupler attached to the end2

3.11.2 Sample Code:

```
(CADS_RCL_ENTGET Barename "number_bars")
Command: !lisplink1
5
```

3.12 CADS_RCL_ FillDims

This function helps to get the actual bar diameter from the type files.

3.12.1 Syntax:

(CADS_RCL_GetActualBarSize TypeOrGrade as string, BarSize as string)

3.12.2 Sample Code:

(CADS_RCL_GetActualBarSize "FE-500" "5")

5.0

3.13 CADS_RCL_GetAllBTypes

This function helps to get all the bar types from the current config. It will store the available bar types in the variable “Lisplink1”

3.13.1 Syntax:

(CADS_RCL_GetAllBTypes)

3.13.2 Sample Code:

(CADS_RCL_GetAllBTypes)

!lisplink1

“FE-500,FE-500D,FE-
415,FE-415D,FE-550,FE-
550D,FE-600,HYS-
500,HYS-415,HYS-
550,MTS,MS-G1,MS-
G2,HYSC-415,HYSC-
500,HYSC-
550,CWS,T,R,”

3.14 CADS_RCL_GetBShapes

This function is used to get all the bar shapes from the current config. It will store the available bar shapes in the variable “Lisplink1”.

3.14.1 Syntax:

(CADS_RCL_GetBShapes)

3.14.2 Sample Code:

(CADS_RCL_GetBShapes)

!lisplink1
"010,011,012,013,014,020,021,022,030,031,032,040,041,042,043,044,050,051,052,053,054,055,060,060A,061,062,063,064,065,066,070,"

3.15 CADS_RCL_GetCFG

This function is used to get the Global configuration value from RebarCAD.

3.15.1 Syntax:

(CADS_RCL_GETCFG ConfigSectionName as string, ConfigName as string)

3.15.2 Sample Code:

(setq ConfigValue (CADS_RCL_GETCFG "bars" "rcbarlay"))

3.15.3 Sample Program:

```
(defun C:GetCfg ()
    (setq BarLayer(CADS_RCL_GETCFG "bars" "rcbarlay")) (princ
    BarLayer)
)
```

3.16 CADS_RCL_GET_CUR_DET_STD

This function is used to get the current detailing standard used in the drawing.

3.16.1 Syntax:

(CADS_RCL_GET_CUR_DET_STD)

3.16.2 Sample Code:

```
(CADS_RCL_GET_CUR_DET_STD)
```

3.16.3 Sample Program:

```
(defun C:Get_current_detailing_standard ()
        (setq detailingStd (CADS_RCL_GET_CUR_DET_STD))
        (princ detailingStd)
        (princ)
)
```

3.17 CADS_RCL_GET_CUR_DET_STD_FILE

This function is used to get the current detailing standard used in the drawing with the filename and location.

3.17.1 Syntax:

```
(CADS_RCL_GET_CUR_DET_STD_FILE)
```

3.17.2 Sample Code:

```
(CADS_RCL_GET_CUR_DET_STD_FILE)
```

3.17.3 Sample Program:

```
(defun C:Get_current_detailing_standard_file ()
        (setq detailingStdFile (CADS_RCL_GET_CUR_DET_STD_FILE))
        (princ detailingStdFile)
        (princ)
)
```

3.18 CADS_RCL_GET_CUR_RC_VER

This function is used to get the current version of RebarCAD.

3.18.1 Syntax:

(CADS_RCL_GET_CUR_RC_VER)

3.18.2 Sample Code:

(CADS_RCL_GET_CUR_RC_VER)

3.18.3 Sample Program:

```
(defun C:Get_Current_RC_Version ()
        (setq curRCVer (CADS_RCL_GET_CUR_RC_VER))
        (princ curRCVer)
        (princ)
)
```

3.19 CADS_RCL_GetDimStrs

This is used to get a list of all the Leg dimension strings for the current Configuration.

3.19.1 Syntax:

(CADS_RCL_GetDimStrs)

3.19.2 Sample Code:

(CADS_RCL_GetDimStrs)

```
!lisplink1
"A,B,C,D,E,F,G,H,I,J,"
```

3.20 CADS_RCL_GetVar

This helps to get the values of the variables specified. It will store the value in the variable called Lisplink1.

3.20.1 Syntax:

(CADS_RCL_GetVar VariableName as String)

3.20.2 Sample Code:

```
(cads_rcl_getvar "profile_mode")
!lisplink1 0
```

3.21 CADS_RCL_IND_BAR_RUN

This function helps to draw a single bar run.

3.21.1 Syntax:

(CADS_RCL_IND_BAR_RUN ShapeCode as string, StartPoint as point, EndPoint as point, CenterToCenterDistance as real, quantity as integer)

3.21.2 Sample Code:

```
(CADS_RCL_IND_BAR_RUN "20" point1 (polar point1 (dtr 0) 100) 0 1)
```

3.21.3 Sample Program:

```
(defun C:DrawRun ()
  (CADS_RCL_ACAD_TOGGLE)
  (setq BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
        Shapecode (getstring "\nEnter the shape Code :")
  ) (CADS_RCL_SET_MEMTITLE)
  (CADS_RCL_SETTYPEDIAM "R" "16")
  (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
  (CADS_RCL_FillDims "1000,2000")
```

```
    (CADS_RCL_DRAW_BAR_ND Shapecode BarInsertionPoint 0.0 "L" "Side") (setq
    Barename (cdr (car (entget (entlast)))))

    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_NEWview Barename)

    (setq RunPoint (polar BarInsertionPoint (dtr 90) 300))

    (CADS_RCL_IND_BAR_RUN Shapecode RunPoint (polar RunPoint (dtr 0) 100) 0 1)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.22 CADS_RCL_LABEL_BAR

This function is used to draw the label for an existing bar.

3.22.1 Syntax:

(CADS_RCL_LABEL_BAR BarEname as entity name, Labelpoint as point, Rotation angle as real, Notes as string)

3.22.2 Sample Code:

;;To get the entity name of the last object drawn.

```
(setq BarEname (cdr (car (entget (entlast)))))
(CADS_RCL_LABEL_BAR BarEname '(0 0) 0.0)
```

3.22.3 Sample Program:

```
(defun C:DrawBar_with_label ()
    (CADS_RCL_ACAD_TOGGLE)

    (setq BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
          Shapecode (getstring "\nEnter the shape Code :") Dimension
          (getreal "\nEnter the dimension: ")

    ) (CADS_RCL_SET_MEMTITLE)

    (CADS_RCL_SETTYPEDIAM "R" "16")

    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No") (CADS_RCL_DRAW_BAR shapecode
    BarInsertionPoint 0.0 "R" "Plan" dimension) (setq Barename (cdr (car (entget (entlast)))))

    (CADS_RCL_NEWSET Barename)

    (setq Labelpoint (getpoint "\nPick the Label Point : "))
    (CADS_RCL_NEWview Barename) (CADS_RCL_LABEL_BAR
    Barename Labelpoint 0.0) (CADS_RCL_ACAD_TOGGLE)
)
```

3.22.4 Sample Program:

```
(defun C:DrawBar_with_label () (CADS_RCL_ACAD_TOGGLE)
        (setq
                BarInsertionPoint (getpoint "\nPick the bar insertion point : ") Shapecode (getstring "\nEnter the shape Code :")
                Dimension (getreal "\nEnter the dimension: ")
                BarType (getstring "\nEnter the bar type :")
                BarSize (getstring "\nEnter the bar size :")
        )
        (CADS_RCL_SET_MEMTITLE)
        (CADS_RCL_SETTYPEDIAM BarType BarSize)
        (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
        (CADS_RCL_DRAW_BAR shapecode BarInsertionPoint 0.0 "R" "Plan" dimension)
        (setq Barename (cdr (car (entget (entlast)))))
        (CADS_RCL_NEWSET Barename)
        (setq Labelpoint (getpoint "\nPick the Label Point : "))
        (CADS_RCL_NEWview Barename)
        (CADS_RCL_LABEL_BAR Barename Labelpoint 0.0 "Test notes")
        (CADS_RCL_ACAD_TOGGLE)
)
```

3.23 CADS_RCL_NEWLABEL_EXTRACTMARK

This function is used to draw a new label, by extracting the data from an existing bar on the drawing.

3.23.1 Syntax:

(CADS_RCL_LABEL_BAR BarMark as string, Labelpoint as point, Rotation angle as real, Multiplier as integer, Quantity as integer)

3.23.2 Sample Program:

```
(defun C:Extract_Bar_Draw_Label () (CADS_RCL_ACAD_TOGGLE)
        (setq
                BarInsertionPoint (getpoint "\nPick he label insertion point : ") Shapecode (getstring "\nEnter the shape Code :")
                Dimension (getreal "\nEnter the dimension: ")
```

```
        )
        (setq BarMark (getstring "\nEnter the bar mark:"))
        (CADS_RCL_NEWLABEL_EXTRACTMARK BarMark BarInsertionPoint 0 5 6)
        (setq Barname (cdr (car (entget (entlast)))))
        (CADS_RCL_NEWSET Barname)
        (CADS_RCL_ACAD_TOGGLE)
)
```

3.24 CADS_RCL_LABELTEXT

This function helps to draw a text object on the screen.

It will draw in the label layer and will take the text default values of the Sample text & insertion point

3.24.1 Syntax:

(CADS_RCL_LABELTEXT Sample Text as string, Insertion point as point)

3.24.2 Sample Code:

```
(CADS_RCL_LABELTEXT "Top Bars" point1)
```

3.24.3 Sample Program:

```
(defun C:DrawLabelText ()
        (CADS_RCL_ACAD_TOGGLE)
        (setq InsertionPoint (getpoint "\nPick the Text insertion point : ")
              SampleText (getstring "\nEnter the sample Text:")
        )
        (CADS_RCL_LABELTEXT SampleText InsertionPoint) (CADS_RCL_ACAD_TOGGLE)
)
```

3.25 CADS_RCL_LEADER

This function is used to draw a leader for the label.

3.25.1 Syntax:

(CADS_RCL_LEADER LabelEname as entity name Arrowpoint as point)

3.25.2 Sample Code:

;;To get the entity name of the last object drawn. (setq ename (cdr (car (entget (entlast))))) (CADS_RCL_LEADER ename p1)

3.25.3 Sample Program:

```
(defun C:DrawBar_with_Leader_label ()
    (CADS_RCL_ACAD_TOGGLE)
    (setq BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
        Shapecode (getstring "\nEnter the shape Code :")
        Dimension (getreal "\nEnter the dimension: ")
    ) (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETTYPEDIAM "R" "16")
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No") (CADS_RCL_DRAW_BAR
    shapecode BarInsertionPoint 0.0 "R" "Plan" dimension)
    (setq Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (setq Labelpoint (getpoint "\nPick the Label Point : "))
    (CADS_RCL_NEWview Barename) (CADS_RCL_LABEL_BAR
    Barename Labelpoint 0.0) (setq Labelename (cdr (car (entget
    (entlast)))))

    (setq LeaderPoint (polar BarInsertionPoint (dtr 0) 100))
    (CADS_RCL_NEWview Barename)
    ;;entname of the label and leader starting point
    (CADS_RCL_LEADER Labelename LeaderPoint)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.26 CADS_RCL_LINE_TAG

This is the function to draw the tick and tag on only one side of the bar.

3.26.1 Syntax:

(CADS_RCL_LINE_TAG BarEname as ExistingbarEntityName, ArrowHeadStartingPoint as point, ArrowLineEndPoint as point, SideFlag as integer)

Side flag can be either 1 or 0.

3.26.2 Sample Code:

```
(setq Barename (cdr (car (entget (car (entsel))))))
(CADS_RCL_LINE_TAG Barename p1 p2 0)
```

3.26.3 Sample Program:

```
(defun c:Drawtag ()
   (CADS_RCL_ACAD_TOGGLE)
   (setq Barename (cdr (car (entget (car (entsel "Select an Existing Bar Object : "))))))
   (CADS_RCL_BAR_ENDS Barename )
   (setq point1 (polar lisplink1 (dtr 270) 200)) (setq
   point2 (polar lisplink1 (dtr 0) 200))
   (CADS_RCL_NEWset Barename)
   (CADS_RCL_NEWview Barename)
   (CADS_RCL_LINE_TAG Barename point1 point2 0)
   (CADS_RCL_LINE_TAG Barename point1 point2 1)
   (CADS_RCL_ACAD_TOGGLE)
)
```

3.27 CADS_RCL_NEWLABEL

This function is used to draw a new label.

3.27.1 Syntax:

(CADS_RCL_NEWLABEL Shapecode as string, LabelInsertion as point, AngleOfLabel in real (value should be in radian), Multiplier as integer, CenterToCenterdistance as real, NoOfBars as integer, Notes as string)

3.27.2 Sample Code:

```
(CADS_RCL_NEWLABEL "20" point1 (dtr 90) 5 100.0 6 "notes")
```

3.27.3 Sample Program:

```
(defun c:NewLabel ()
    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_SETTYPEDIAM "T" "20")
    (setq LabelInsertionpoint (getpoint "\nEnter a insertion point : ")
          ShapeCode (getstring "\nEnter the shape code : ")
    )
    (CADS_RCL_FillDims "1000,2000")
    (CADS_RCL_NEWLABEL ShapeCode LabelInsertionpoint (dtr 90) 5 100.0 6 "notes")
    (setq Barename (cdr (car (entget(entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.28 CADS_RCL_NEWLABEL_MARK

This function is used to draw a new label for a given bar mark.

3.28.1 Syntax:

(CADS_RCL_NEWLABEL_MARK BarMark as string, ShapeCode as string, LabelInsertion as point, Type as string, Size as string, , AngleOfLabel in real (value should be in radian), Multiplier as integer, CenterToCenterdistance as real, NoOfBars as integer, Notes as string)

3.28.2 Sample Code:

(CADS_RCL_NEWLABEL_MARK BarMark ShapeCode refPoint 0 5 100.0 6 "notes")

3.28.3 Sample Program:

```
(defun C:DrawBar_with_label_BarMark ()
        (CADS_RCL_ACAD_TOGGLE)
        (CADS_RCL_SET_MEMTITLE)
        (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
        (CADS_RCL_SETTYPEDIAM "A615/60" "#5")
        (setq refPoint (list 100.0 1008.0 0.0))
        (setq ShapeCode "16a")
        (setq BarMark "501")
```

```
        (setq dimstring (strcat "10" "," "49.57" "," "74.29"))
        (CADS_RCL_FillDims dimstring)
        (CADS_RCL_NEWLABEL_MARK BarMark ShapeCode refPoint 0 5 100.0 6 "notes")
        (setq Barename (cdr (car (entget (entlast)))))
        (alert BarMark)
        (CADS_RCL_NEWSET Barename)
        (CADS_RCL_ACAD_TOGGLE)
)
```

3.29 CADS_RCL_NEWSET

This is used to draw a new set. To convert the RC Object into a New Set.

3.29.1 Syntax:

(CADS_RCL_NEWSET RcEntityName as entity name)

3.29.2 Sample Code:

```
(CADS_RCL_DRAW_BAR “20” ‘(0 0) 0.0 "R" "Plan" 2000)
(setq Barename (cdr (car (entget (entlast)))))
(CADS_RCL_NEWSET Barename)
```

3.29.3 Sample Program:

```
(defun C:DrawBar ()
    (CADS_RCL_ACAD_TOGGLE)
    (setq BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
          Shapecode (getstring "\nEnter the shape Code :") Dimension
          (getreal "\nEnter the dimension: ")
    ) (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETTYPEDIAM "R" "16")
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No") (CADS_RCL_DRAW_BAR
    shapecode BarInsertionPoint 0.0 "R" "Plan" dimension) (setq Barename (cdr (car (entget
    (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.30 CADS_RCL_NEWVIEW

This is used to add a view for an existing RC object. Before drawing the view of an existing set, the entity name of the existing set should be passed.

3.30.1 Syntax:

(CADS_RCL_NEWview RcEntityName as entity name)

3.30.2 Sample Code:

```
(setq Barename (cdr (car (entget (entlast)))))
(CADS_RCL_NEWview Barename)
(CADS_RCL_DRAW_BAR "37" (polar point1 (dtr 270) 300) 0.0 "L" "Plan" 1000)
```

3.30.3 Sample Program:

```
(defun C:DrawBar_with_label ()
    (CADS_RCL_ACAD_TOGGLE)
    (setq BarInsertionPoint (getpoint "\nPick the bar insertion point : ")
        Shapecode (getstring "\nEnter the shape Code :") Dimension
        (getreal "\nEnter the dimension: ")
    ) (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETTYPEDIAM "R" "16")
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No") (CADS_RCL_DRAW_BAR shapecode
    BarInsertionPoint 0.0 "R" "Plan" dimension) (setq Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (setq Labelpoint (getpoint "\nPick the Label Point : "))
    (CADS_RCL_NEWview Barename) (CADS_RCL_LABEL_BAR
    Barename Labelpoint 0.0) (CADS_RCL_ACAD_TOGGLE)
)
```

3.31 CADS_RCL_RANGE

This function helps to draw the range.

Currently it supports only single indicator ranges.

3.31.1 Syntax:

(CADS_RCL_RANGE Shapecode as string, RangeStartPoint as point, RangeEndPoint as point, BarInsertionPoint as point, BarAngle as real, Handing as string, View as string, CenterToCenterDistance as real, NumberOfBars as integers, Dimensions as real)

3.31.2 Sample Code:

```
(CADS_RCL_RANGE "20" RangeStartpt RangeEndpt BarInspt 90.0 "R" "Side" 380.0 10
1000)
```

3.31.3 Sample Program:

```
(defun c:DrawRange ()
    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_SETTYPEDIAM "T" "20")
    (setq BarInspt (getpoint "\nPick Bar Insertion pt: ")
        RangeStartpt (getpoint "\nPick Range Start Pt: ")
        RangeEndpt(getpoint RangeStartpt "\nPick Range End pt: ")
    )
    (CADS_RCL_RANGE "20" RangeStartpt RangeEndpt BarInspt 90.0 "R" "Side" 380.0 10
    1000)
    (setq Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.32 CADS_RCL_RANGE_ND

This function is used to draw ranges.

Currently it supports only single indicator ranges.

You need to use the CADS_RCL_FillDims to set the leg dimensions.

3.32.1 Syntax:

(CADS_RCL_RANGE_ND Shapecode as string, RangeStartPoint as point, RangeEndPoint as point, BbarInsertionPoint as point, BarAngle as real, Handing as string, View as string, CenterToCenterDistance as real, NumberOfBars as integers, Dimesions as real)

3.32.2 Sample Code:

(CADS_RCL_RANGE_ND "20" RangeStartpt RangeEndpt BarInspt 90.0 "R" "Side" 380.0 100.0)

3.32.3 Sample Program:

```
(defun c:RangeWithFillDims ()
    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_SETTYPEDIAM "T" "20")
    (setq BarInspt (getpoint "\nPick Bar Insertion pt: ")
          RangeStartpt (getpoint "\nPick Range Start Pt: ")
          RangeEndpt(getpoint RangeStartpt "\nPick Range End pt: ")
     )
    (CADS_RCL_FillDims "1000")
    (CADS_RCL_RANGE_ND "20" RangeStartpt RangeEndpt BarInspt 90.0 "R" "Side" 380.0
    10 0.0)
    (setq Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.33 CADS_RCL_RANGE_SFD

This function helps to draw ranges.

Currently it supports only single indicator ranges.

3.33.1 Syntax:

(CADS_RCL_RANGE_SFD Shapecode as string, RangeStartPoint as point, RangeEndPoint as point, BbarInsertionPoint as point, BarAngle as real, Handing as string, View as string, CenterToCenterDistance as real, NumberOfBars as integers, Dimesions as real)

3.33.2 Sample Code:

(CADS_RCL_RANGE_SFD "62" RangeStartpt RangeEndpt BarInspt 0.0 "R" "Plan" 150.0 9
1079 643 1223 90.0)

3.33.3 Sample Program:

```
(defun c:DrawRangeSFD ()
    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_SETTYPEDIAM "T" "20")
    (setq BarInspt (getpoint "\nPick Bar Insertion pt: ")
          RangeStartpt (getpoint "\nPick Range Start Pt: ")
          RangeEndpt(getpoint RangeStartpt "\nPick Range End pt: ")
    )
    (CADS_RCL_RANGE_SFD "62" RangeStartpt RangeEndpt BarInspt 0.0 "R" "Plan" 150.0 9 1079
    643 1223 90.0)
    (setq Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.34 CADS_RCL_SET_LAYER

This is used to set the layer based on the layer index.

3.34.1 Syntax:

(CADS_RCL_SET_LAYER layer index as integer)

For example if the layer index is 2, it will take the “RcBarLay" from the current config and sets the layer as current. The layer index ranges from 1 – 10. Details of layer index follows

1. LabelLay ("Label", "LabelLay"),
2. RcBarLay ("Bars", "RcBarLay"),
3. RcBarSecLay ("Bars", "RcBarSecLay"),
4. OutlineLay ("Outlines", "OutlineLay"),
5. CoverLay ("Outlines", "CoverLay"),
6. MarkerLay ("SectionMarkers", "MarkerLay"),
7. MarkerTxtLay ("SectionMarkers", "MarkerTxtLay"),
8. LeaderLay ("RcLeader", "LeaderLay"),
9. DrawSchedLay ("ScheduleConfig", "DrawSchedLay"),
10. Layer ("BarRefs", "Layer");

3.34.2 Sample Code:

```
(CADS_RCL_SET_LAYER 2)
```

3.35 CADS_RCL_SET_MEMTITLE

This function displays the cads_rc membertitle dialog.

3.35.1 Syntax:

(CADS_RCL_SET_MEMTITLE)

3.35.2 Sample Code:

(CADS_RCL_SET_MEMTITLE)

3.35.3 Sample Program:

```
(defun C:DisplayMemberTitle ()

    (CADS_RCL_SET_MEMTITLE)
)
```

3.36 CADS_RCL_CREATE_MEMBER

This function creates a new member title with the required quantity. The function also provides option to set the newly created member as the current member title in the drawing.

3.36.1 Syntax:

(CADS_RCL_CREATE_MEMBER MemberTitle as string, Description as string, Quantity as integer, SetCurrent as string)

3.36.2 Sample Code:

(CADS_RCL_CREATE_MEMBER "member" "description" 1 "false")

3.36.3 Sample Program:

```
(defun C:CreateMemberTitle ()
     (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_CREATE_MEMBER "member" "description" 1 "false")
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.37 CADS_RCL_CREATE_RELEASE

This function creates a new release number with the required description, colour code and sub release. The function also provides option to set the newly created release as the current release title in the drawing.

3.37.1 Syntax:

CADS_RCL_CREATE_ RELEASE ReleaseName as string, Description as string, ColorIndex as integer, SetCurrent as string, SubRelease as string)

3.37.2 Sample Code:

```
(CADS_RCL_CREATE_RELEASE "Release0")
(CADS_RCL_CREATE_RELEASE "Release1" "Release Description 1")
(CADS_RCL_CREATE_RELEASE "Release2" "Release Description 2" 0)
(CADS_RCL_CREATE_RELEASE "Release3" "Release Description 3" 1 "YES")
(CADS_RCL_CREATE_RELEASE "Release4" "Release Description 4" 2 "NO" "SR4")
```

3.37.3 Sample Program:

```
(defun C:CreateReleaseNumber ()
    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_CREATE_RELEASE "Release1" "Release Description 1" 1 "YES" "SR1")
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.38 CADS_RCL_SetBType

This is used to set a bar type before drawing RC Objects.

3.38.1 Syntax:

(CADS_RCL_SetBType type as string)

3.38.2 Sample Code:

```
(CADS_RCL_SetBType "T")
```

3.39 CADS_RCL_SETCFG

This function is used to set the Global configuration value to RebarCAD.

3.39.1 Syntax:

(CADS_RCL_SETCFG Config section name as string, config name as string, config value as string)

3.39.2 Sample Code:

```
(CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
```

3.39.3 Sample Program:

```
(defun C:SetCfg ()
    (CADS_RCL_SETCFG "bars" "AskForBmarkinput" "No")
)
```

3.40 CADS_RCL_SETTYPEDIAM

This function is used to set the type and diameter of the bar. This function should be set before drawing the bar.

3.40.1 Syntax:

(CADS_RCL_SETTYPEDIAM Type as string, Diameter as string)

3.40.2 Sample Code:

```
(CADS_RCL_SETTYPEDIAM "R" "16")
```

3.41 CADS_RCL_STAPRANGE

This function is used to draw the tapered range.

3.41.1 Syntax:

(CADS_RCL_ STAPRANGE ShapeCode as string, RangeStartPoint as point, RangeEndPoint as point, BarInsertionPoint as point, BarAngle as real, Handing as string, View as string, CenterToCenter as real, NumberOfBars as integer, SketchBarAdim as real, SketchBarBdim as real, SketchBarCdim as

real, SketchBarDdim as real, SketchbarEdim as real, SketchbarFdim as real, SketchBarGdim as real, SketchBarHdim as real, SketchBarIdim as real, SketchBarJdim as real, FirstBarAdim as real, FirstBarBdim as real, FirstBarCdim as real, FirstBarDdim as real, FirstBarEdim as real, FirstBarFdim as real, FirstBarGdim as real, FirstBarHdim as real, FirstBarIdim as real, FirstBarJdim as real, SecondBarAdim as real, SecondBarBdim as real, SecondBarCdim as real, SecondBarDdim as real, SecondBarEdim as real, SecondBarFdim as real, SecondBarGdim as real, SecondBarHdim as real, SecondBarIdim as real, SecondBarJdim as real)

3.41.2 Sample Code

(CADS_RCL_STAPRANGE "20" RangeStartpt RangeEndpt BarInspt 90 "R" "Side" 150.0 9 3000 0.0 0.0 0.0 0.0 0.0 0.0

0.0 0.0 0.0 1000.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 2000.0

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0)

3.41.3 Sample Program:

```
(defun c:DrawTaperRange ()
    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_SETTYPEDIAM "T" "20")
    (setq BarInspt (getpoint "\nPick Bar Insertion pt: ")
        RangeStartpt (getpoint "\nPick Range Start Pt: ")
        RangeEndpt(getpoint RangeStartpt "\nPick Range End pt: ")
    )
    (CADS_RCL_STAPRANGE "20" RangeStartpt RangeEndpt BarInspt 90 "R" "Side" 150.0 9 3000
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 1000.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 2000.0 0.0 0.0 0.0 0.0 0.0  0.0 0.0 0.0 0.0 )
    (setq Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.42 CADS_RCL_TICK_TAG

This function is used to draw the tick and tag for an existing bar.

3.42.1 Syntax:

(CADS_RCL_TICK_TAG BarEname as entity name, SidePoint as point, Swapflag as integer)

Swapflag should be either 1 or 0

3.42.2 Sample Code:

(CADS_RCL_TICK_TAG Barename point1 0)

3.42.3 Sample Program:

```
(defun c:DrawTickAndTAG ()

    (CADS_RCL_ACAD_TOGGLE)
    (CADS_RCL_SET_MEMTITLE)
    (CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")
    (CADS_RCL_SETTYPEDIAM "T" "20")
    (setq BarInsertionPoint (getpoint "\nEnter a insertion point : "))
    (CADS_RCL_DRAW_BAR "20" BarInsertionPoint 0.0 "L" "Side" 1000 ) (setq
    Barename (cdr (car (entget (entlast)))))
    (CADS_RCL_NEWSET Barename)
    (setq TagPoint (getpoint "\nEnter a side point : "))
    (CADS_RCL_TICK_TAG Barename TagPoint 0 )
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.43 CADS_RCL_TOGDRAWMODE

This function helps to control the style of the bar.

3.43.1 Syntax:

(CADS_RCL_TOGDRAWMODE BarStyle as integer)

BarStyle should be either 1 or 0

If the value is 0 then the bar will be drawn in the centre mode. If the value is 1 then the bar will be drawn in the profile mode.

3.43.2 Sample Code:

(CADS_RCL_TOGDRAWMODE 1)

3.44 CADS_RCL_TOGBMPROMPT

This function is used to control the bar mark request prompt.

3.44.1 Syntax:

(CADS_RCL_TOGBMPROMPT BarMarkPrompt as integer)

BarMarkPrompt should be either 1 or 0

If the value is 0 then it will ask you to enter the bar mark.

If the value is 1 then it will not ask you to enter the bar mark.

3.44.2 Sample Code:

```
(CADS_RCL_TOGBMPROMPT 1)
```

3.45 CADS_RCL_UPCCNUM

This function helps to update the Centre to Centre spacing, number of bars & multiplier

3.45.1 Syntax:

(CADS_RCL_UPCCNUM BarEname as EntityName, CCSpacing as real, NumberOfBars as integer, multiplier as integer)

3.45.2 Sample Code:

```
(CADS_RCL_UPCCNUM Barename 100 10 5)
```

3.45.3 Sample Program:

```
(defun C:UpCCNum ()
    (CADS_RCL_ACAD_TOGGLE)
    (setq Barename (cdr (car (entget (car (entsel "Select an existing Rc Object :")))))
          CCValue (getreal "\nEnter new C/C Spacing ")
          NoBars (getint "\nEnter the number of bars :") Multi
          (getint "\nEnter the number of Multiplier: ")
    )
    (CADS_RCL_UPCCNUM Barename CCValue nobars multi)
    (CADS_RCL_ACAD_TOGGLE)
)
```

3.46 CADS_RCL_UPCCNUMHIDMULT

This function helps to update the Centre to Centre spacing, number of bars, multiplier & hidden multiplier.

3.46.1 Syntax:

(CADS_RCL_UPCCNUMHIDMULT BarEname as EntityName, CCSpacing as real, NumberOfBars as integer, Multiplier as integer, HiddenMultiplier as integer)

3.46.2 Sample Code:

(CADS_RCL_UPCCNUMHIDMULT Barename 100 10 5 2)

3.46.3 Sample Program:

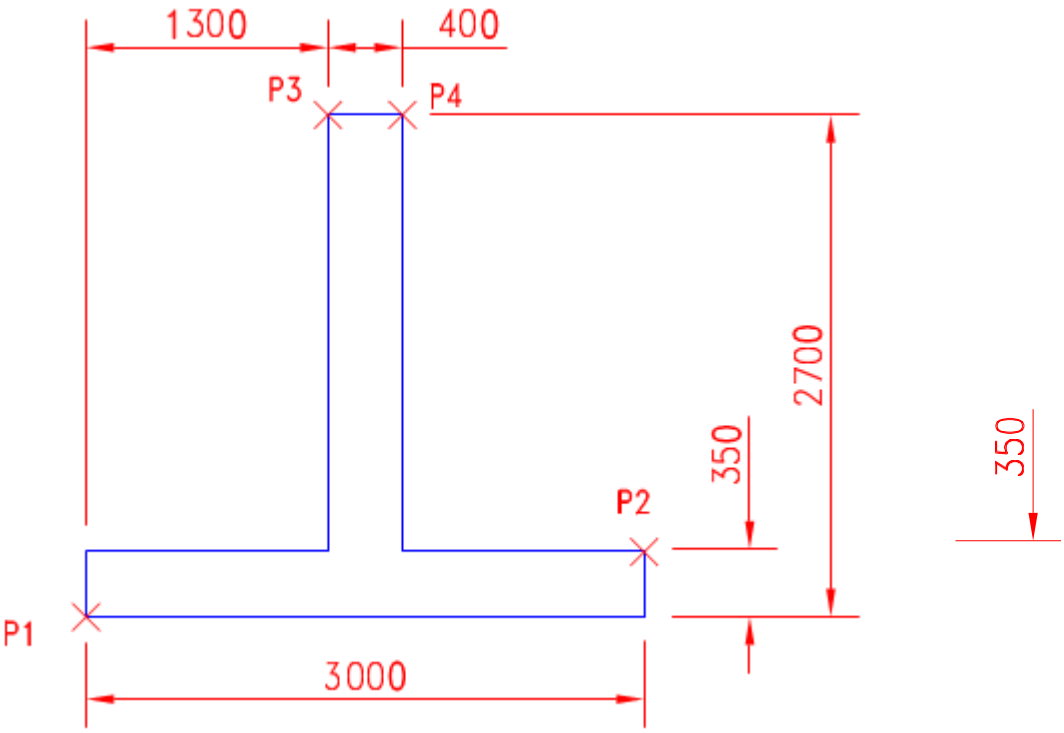
```
(defun C:UpCCNumHidMult ()
    (CADS_RCL_ACAD_TOGGLE)
    (setq Barename (cdr (car (entget (car (entsel "Select an existing Rc Object :")))))
          CCValue (getreal "\nEnter new C/C Spacing ")
          NoBars (getint "\nEnter the number of bars: ")
          multi (getint "\nEnter the number of Multiplier: ")
          hiddenmulti (getint "\nEnter the Hidden Multiplier: ")
)
    (CADS_RCL_UPCCNUMHIDMULT Barename CCValue nobars multi
    hiddenmulti)

    (CADS_RCL_ACAD_TOGGLE)
)
```

Limitation of the Exposed function.

- Double click edit is not possible for the RC objects which are drawn using the Exposed function.
- After creating the RC objects through the Lisp program, use “cads_rc_redraw" command to get the double click edit options.

The sample Program for the footing follows:



3.46.4 Procedure to draw the reinforcement:

1. First draw the outlines with the required dimension.
2. Create a folder called LispDemo in C:\
3. Create a lisp file called Footing.lsp in C:\LispDemo folder
4. Copy and paste the code mentioned below to the Footing.lsp
5. Save it and close.
6. Create a text file called Reinforcement.txt in the same C:\LispDemo Folder
7. Copy the reinforcement text file details (available in the footing function given below) to the reinforcement.txt file.
8. Save it and close.
9. In AutoCAD use the Appload command to load the Footing.lsp
10. Type **Footing** in the command prompt. It will ask you to pick the Lower Left corner of the base & upper right corner of the base. Pick points p1 & p2 as mentioned in the drawing.
11. For Wall left corner and right corner pick p3 & P4 as mentioned in the drawing.
12. You can see the reinforcement drawn on the screen.
13. Repeat steps 10 & 11 for different footing outlines.

3.46.5 Sample coding to draw the reinforcement for the footing.

```
;; function to draw the footing reinforcement details
(defun c:Footing ()
        ;;std function call at the beginning and end of the functions
        ;;it is a mandatory function
        (CADS_RCL_ACAD_TOGGLE)
```

```
;;function call to display and set the member title
(CADS_RCL_SET_MEMTITLE)

;;function to set the configuration

;;so that it won't ask you to specify the bar mark each time
(CADS_RCL_SETCFG "Bars" "AskForBmarkinput" "No")

;;std function call
(CADS_RCL_ACAD_TOGGLE)

;;to store the old osnap mode values
(setq old_osmode (getvar "osmode"))

;;to set the osnap for end & intersection
(command "osmode" 33)

;;getting the points from the user

;; Pick the points p1, p2, p3 & p4 respectively

(setq   p1 (getpoint "\nPick lower left corner of the base: ")
        p2 (getpoint"\nPick upper right corner of the base: ")
        p3 (getpoint"\nPick wall left point:")

        p4 (getpoint"\nPick wall right point: ")

)

;;to revert the osmde

(command "osmode" old_osmode)

;;function call to set the dimension
(setdimensions)

;; open a reinforcement detail file

(setq fp (open "C:/LispDemo/reinforcement.txt" "r"))

;;read the first line of the file
(read-line fp)

;;getting the cover value from the file
(setq cover (atoi (read-line fp)))

(read-line fp)

;;getting the lap value from the file
(setq Lap (atoi (read-line fp)))

;;getting the diameter, grade and ccspacing for the bottombar
(read-line fp)

(setq diamandgrade (read-line fp))
(setq ccspacing (atof (read-line fp)))

;; function call to draw the bottom bar & Run
(DrawBottomBar)

;;getting the diameter, grade and ccspacing for the dowelbar
(read-line fp)

(setq diamandgrade (read-line fp))
(read-line fp)
```

```
        ;; function call to draw the drawdowel
        (DrawDowel)
        ;;getting the diameter, grade and ccspacing for the Linkbars
        (read-line fp)
        (setq diamandgrade (read-line fp)
              ccspacing (atof (read-line fp))
        )
        ;; function call to draw the drawlinks
        (DrawLinks)
        ;;close the reinforcement.txt file
        (close fp)
        ;; redrawing all the bars drawn so that you can utilize the double click option for later
        ;;editing
        (command "cads_rc_redraw" "all" "")
) ;;Main function completed
;;function to draw link bars
(defun Drawlinks (/ Barename)
        ;;std function call
        (CADS_RCL_ACAD_TOGGLE)
        ;;function call to set the type and grade
        (settypeandgrade diamandgrade)
        ;; to calculate the no of links
        (setq nooflinks (fix (/ (- wallheight Foun_Depth ccspacing) ccspacing)))
        ;;to set the Leg dimension , insertion point for the bar and ranges
        (setq inspt (polar p2 (dtr 180) (+ (/ (- foun_width wallthick) 2.0) cover)))
        (setq inspt (polar inspt (dtr 90) (* 2 Foun_Depth)))
        (setq dimA (- wallthick cover cover))
        (setq dimstring(rtos dimA))
        (setq RangeStartpt (list (- (car inspt) (/ wallthick 2.0)) (+ (cadr p2) (/ ccspacing 2.0))))
        (setq RangeEndpt (polar RangeStartpt (dtr 90) (- wallheight Foun_Depth ccspacing)))
        ;;to draw the Links
        (drawrange dimstring "51" RangeStartpt RangeEndpt inspt 0.0 "L" "Plan" ccspacing)
        ;; std function call
        (CADS_RCL_ACAD_TOGGLE)
) ;; draw links function completed
;;function to draw dowelbar
(defun DrawDowel(/ dimA dimB Barename)
        ;;std function call
        (CADS_RCL_ACAD_TOGGLE)
        ;;function call to set the type and grade
```

```
        (settypeandgrade diamandgrade)

        ;; to set the insertion point and Leg dimension for the Left side bar
        (setq dowelinspt (polar Rangeinspt (dtr 90) (* (atof diameter) 3.0)))
        (setq inspt (list (+ (car p3) cover (atof diameter)) (cadr dowelinspt)))
        (setq dimA (* lap (atof diameter))

            dimB (distance inspt (list (car inspt) (- (cadr p3) (/ ccspacing 2))))

        )
        (setq dimstring (strcat (rtos dimA) "," (rtos dimB) ))

        ;; function call to draw the Left side dowel bar
        (drawbar dimstring "11" inspt 90.0 "R" "Side")

        ;; to set the insertion point and the Leg dimesion for the Right Side bar
        (setq dowelinspt (polar dowelinspt 0 (- WallThick cover cover) ))

        (setq inspt (list (- (car p4) cover (atof diameter)) (cadr dowelinspt)))

        ;; function call to draw the right side dowel bar
        (drawbar dimstring "11" inspt 270.0 "L" "Side")

        ;;std          function          call
        (CADS_RCL_ACAD_TOGGLE)
)
;;function to draw bottom bars
(defun DrawBottomBar(/ dimA dimB dimC Barename)

        ;;       std       function       call
        (CADS_RCL_ACAD_TOGGLE)

        ;;function call to set the type and grade
        (settypeandgrade diamandgrade)

        ;; to set the insertion point and Leg dimension for the Bottom bar
        (setq inspt (list (+ (car p1) cover) (+ (cadr p1) cover))

            dimA (distance inspt (polar inspt (dtr 90) (- Foun_Depth cover cover)))
            dimB  (- Foun_Width cover cover)

            dimC dimA

        )
        (setq dimstring (strcat (rtos dimA) "," (rtos dimB) "," (rtos dimC)))

        ;; function call to draw the bottom bar
        (drawbar dimstring "21" inspt 0.0 "R" "Side")

        ;; to draw the Bottom bar run
        (CADS_RCL_NEWview Barename)

        (setq Rangeinspt (polar inspt (dtr 90) (* (atof diameter) 1.0)))

        (CADS_RCL_BAR_RUN "21" Rangeinspt (polar inspt (dtr 0) (- dimB cover)) ccspacing 0)

        ;;       std       function       call
        (CADS_RCL_ACAD_TOGGLE)

)
;;general routine to draw the range
```

```
(defun drawRange (dimstring shapecode rangestartpt rangeEndpt Barinspt BarAngle Handing
Style ccspcing)
        ;;function call to set the Leg dimensions
        (CADS_RCL_FillDims dimstring)
        ;; to switch off the osnap
        (setq osmod (getvar "osmode"))
        (setvar "osmode" 0)
        ;; function to draw the range
        (CADS_RCL_RANGE_ND shapecode rangestartpt rangeEndpt Barinspt BarAngle
        Handing Style ccspacing 0 0.0)
        (setvar "osmode" osmod)
        (setq Barename (entlast))
        ;; to set the range as a new set
        (CADS_RCL_NEWSET Barename)
)
;;general function to draw a bar
(defun drawbar(dimstring Shapecode inspt Barangle Handing Style)
        ;; function call to set the Leg dimensions
        (CADS_RCL_FillDims dimstring)
        ;;function to draw the bar
        (CADS_RCL_DRAW_BAR_ND shapecode inspt Barangle Handing Style)
        ;; to set the drawn bar as a new set (setq
        Barename (entlast))
        (CADS_RCL_NEWSET Barename)
)
;;function to set the type and grade
(defun settypeandgrade (diamandgrade)
        (setq grade (substr diamandgrade 1 1))
        (setq diameter (substr diamandgrade 3 2))
        (CADS_RCL_SETTYPEDIAM grade diameter)
)
;; function to get the dimension from the user input points
(defun setdimensions ()
        (setq Foun_Width (distance p1 (list (car p2) (cadr p1)))
            Foun_Depth (distance p1 (list (car p1) (cadr p2)))
            WallThick (distance p3 p4)
            wallheight (distance p1 (list (car p1) (cadr p3)))
        )
)
;;function to convert degrees to radians
```

```
(defun dtr(angle)
  (setq rad (* (/ pi 180.0) angle))
)
```

You need to create a folder C:\LispDemo and create a file Reinforcement.txt to set the reinforcement details

Sample reinforcement.txt file:

```
Cover
50
Lap   Value
48
BottomMat Type and grade
T 16
250
Dowel   Bar
T 20
4
Links T
8
200
Bottom bar in the wall
T 12
150
"L"   bar
T 12
200
```

4 Exposed Functions From Supplementary Programs

4.1 CADS_VPML_CREATE_LAYOUT

This function is used to create a new Layout using CADS Viewport Manager.

4.1.1 Syntax:

(CADS_VPML_CREATE_LAYOUT Scale as string, LayoutIndex as String, LayoutName as string, Sheet as string, DrawingSheetNoAttribute as string, DrawingSheetNo as string)

4.1.2 Sample Code

```
(CADS_VPML_CREATE_LAYOUT "1:1/4\"" "1" "Layout2" "24x36" "DRAWINGNO" "D1")
```

4.1.3 Sample Program

```
(defun C:CADS_VPML_Create_Layout ()
        (CADS_RCL_ACAD_TOGGLE)
        (CADS_VPML_CREATE_LAYOUT "1:1/4\"" "1" "Layout2" "24x36" "DRAWINGNO" "D1")
        (CADS_RCL_ACAD_TOGGLE)
)
```

4.2 CADS_VPML_CREATE_VIEWPORT

This function is used to create a new Viewport using CADS Viewport Manager.

4.2.1 Syntax:

(CADS_VPML_CREATE_VIEWPORT LayoutName as string, ViewportName as string, ViewportShape as string)

4.2.2 Sample Code

```
(CADS_VPML_CREATE_VIEWPORT "Layout2" "view1" "1:20" "Rectangular")
```

4.2.3 Sample Program

```
(defun C:CADS_VPML_Create_VIEWPORT ()
        (CADS_RCL_ACAD_TOGGLE)
        (CADS_VPML_SETUP_VIEWPORT_Point "8'33/64\"" "10'16\"" "0'0\"")
        (CADS_VPML_SETUP_VIEWPORT_Point "25'3/32\"" "30'10\"" "0'0\"")
        (CADS_VPML_SETUP_VIEWPORT_Point "0" "0" "0")
        (CADS_VPML_CREATE_VIEWPORT "Layout2" "view1" "1:1/4"" "Rectangular")
        (CADS_RCL_ACAD_TOGGLE)
)
```

4.3 CADS_VPML_GETCFG

This function is used to get the Global configuration value from CADS Viewport Manager.

4.3.1 Syntax:

```
(CADS_VPML_GETCFG  ConfigSectionName as string,   ConfigName as string)
```

4.3.2 Sample Code:

```
(setq ConfigValue  (CADS_VPML_GETCFG "Miscellaneous" "TextMtextScalingOption"))
```

4.3.3 Sample Program:

```
(defun C:GetCfg ()
    (setq AutoScaleTextOption  (CADS_VPML_GETCFG "Miscellaneous" "TextMtextScalingOption"))
    (princ AutoScaleTextOption)
)
```

4.4 CADS_VPML_SETCFG

This function is used to set the Global configuration value in CADS Viewport Manager.

4.4.1 Syntax:

(CADS_VPML_SETCFG ConfigSectionName as string, ConfigName as string, ConfigValue as string)

4.4.2 Sample Code:

(CADS_VPML_SETCFG "Miscellaneous" "IgnoreScalingForNonRCText" "No")

4.4.3 Sample Program:

```
(defun C:SetCfg ()
    (CADS_VPML_SETCFG "Miscellaneous" "IgnoreScalingForNonRCText" "No")
)
```