CFS13 Automation Module

Copyright © 2021 RSG Software, Inc.

Overview

CFS13 is a 32-bit class library which provides an API to the core CFS 13 calculation engine for performing design calculations for cold-formed steel structural members. It is also available as a 64-bit library if required for use in 64-bit applications.

System Requirements

- Windows 32/64-bit operating system
- .NET Framework 4.5
- Development environment supporting COM (i.e., Microsoft Excel VBA) or .NET 4.5 (i.e., Microsoft Visual Studio)
- CFS 13 software installed and licensed. The CFS license may be a Single User License or a Network License, but must include the automation option.

Installation

Run the installer program cfs13dllinst.exe with administrator permissions. This will step you through the installation process. If you do not have the .NET Framework 4.5 installed, you will be prompted to download and install it.

Licensing

The license to use the **CFS13** module is bundled with a CFS license. When you purchase the license, choose the Single User License with Automation or the Network License with Automation. Then use the CFS software to activate the license.

COM Use

Development tools that support COM, such as Microsoft Excel VBA, can reference the **CFS13** module. To add a reference in Excel, open the Visual Basic editor and choose References from the Tools menu. Then select the **CFS13** Automation Module.

.NET Use

Development tools that support .NET 4.5, such as Microsoft Visual Studio 2010 or newer, can natively reference the **CFS13** assembly. To add a reference in Visual Studio, use the Browse feature from the Add Reference window to select the assembly (i.e., C:\Program Files (x86)\RSG Software\CFS13 Module\ CFS13.dll).

Excel VBA Sample Code

Module1

```vba
Option Explicit
Public cfsCalc As CFS13.Calculation

Public Sub Test()
  On Error GoTo TestError
  If cfsCalc Is Nothing Then Set cfsCalc = New CFS13.Calculation
  If Not cfsCalc.HasLicense Then MsgBox "License not available": Exit Sub

  cfsCalc.LoadSection "C:\CFS Files\Test.cfss"
  Dim cfsProp As CFS13.SectionProperties
  Set cfsProp = cfsCalc.GrossProperties()
  MsgBox "Gross Area = " & cfsProp.Area

  Dim cfsParams As New CFS13.MemberParams
  cfsParams.Lx = 96
  cfsParams.Ly = 48
  cfsParams.Lt = 48
  cfsParams.Lm = 96
  cfsParams.Spec = CFS13.Specification.Specification_AISI2018USASD

  Dim cfsForces As New CFS13.SectionForces
  cfsForces.P = 10
  cfsForces.Mx = 20

  Dim cfsCheck As CFS13.MemberUnityCheck
  Set cfsCheck = cfsCalc.MemberCheck(cfsParams, cfsForces)
  MsgBox "Unity Check 1=" & cfsCheck.PMxMy1
  Exit Sub

TestError:
  MsgBox Err.Source & " Error: " & Err.Description
End Sub

Public Function GrossArea(strFilename As String) As Single
  If cfsCalc Is Nothing Then Set cfsCalc = New CFS13.Calculation
  If Not cfsCalc.HasLicense Then Exit Function

  cfsCalc.LoadSection strFilename
  Dim cfsProp As CFS13.SectionProperties
  Set cfsProp = cfsCalc.GrossProperties()
  GrossArea = cfsProp.Area
End Function
```

ThisWorkbook

```vba
Private Sub Workbook_BeforeClose(Cancel As Boolean)
  'Release network license before the workbook closes
  If Not (Module1.cfsCalc Is Nothing) Then cfsCalc.ReleaseLicense
End Sub
```

VB.NET Sample Code

Module1.vb

```vb
Imports RSG.CFS13

Module Module1
    Public cfsCalc As Calculation

    Public Sub Test()
        If cfsCalc Is Nothing Then cfsCalc = New Calculation
        If Not cfsCalc.HasLicense Then Debug.Print("License not available") : Exit Sub

        cfsCalc.LoadSection("C:\CFS Files\Test.cfss")
        Dim cfsProp As SectionProperties = cfsCalc.GrossProperties()
        Debug.Print("Gross Area=" & cfsProp.Area)

        Dim cfsParams As New MemberParams
        cfsParams.Lx = 96
        cfsParams.Ly = 48
        cfsParams.Lt = 48
        cfsParams.Lm = 96
        cfsParams.Spec = Specification.AISI2018USASD

        Dim cfsForces As New SectionForces
        cfsForces.P = 10
        cfsForces.Mx = 20

        Dim cfsCheck As MemberUnityCheck = cfsCalc.MemberCheck(cfsParams, cfsForces)
        Debug.Print("Unity Check 1=" & cfsCheck.PMxMy1)
    End Sub

    Public Function GrossArea(strFilename As String) As Single
        If cfsCalc Is Nothing Then cfsCalc = New Calculation
        If Not cfsCalc.HasLicense Then Return 0
        cfsCalc.LoadSection(strFilename)
        Return cfsCalc.GrossProperties().Area
    End Function

End Module
```

ApplicationEvents.vb

```vb
Namespace My
    Partial Friend Class MyApplication
        Private Sub MyApplication_Shutdown(sender As Object, e As EventArgs) _
            Handles Me.Shutdown
            'Release network license before the application closes
            If Module1.cfsCalc IsNot Nothing Then Module1.cfsCalc.ReleaseLicense()
        End Sub
    End Class
End Namespace
```

API Documentation

Namespace: RSG.CFS13

Calculation Class

Initialization Methods and Properties

Sub **DefineSection**(*SctData* As [SectionData](#))

Defines a CFS section to be used in subsequent calculations

SctData: Section data object which defines the type of section, dimensions, and material properties

Sub **LoadSection**(*Filename* As String, [*Fy* As Single], [*Fu* As Single], [*ColdWork* As Boolean], [*Reserve* As Boolean], [*UseDSM* As Boolean])

Loads a CFS section file to be used in subsequent calculations

Filename: Name of a CFS section file (*.sct or *.cfss) or the name of a section in a CFS section library (*.scl or *.cfsl). Example: CFS Files\SSMA.cfsl|U-Channels\250U050-54.cfss

Fy: Optional. Yield stress override (ksi). If omitted, the yield stress stored in the section file will be used.

Fu: Optional. Tensile strength override (ksi). If omitted, the tensile strength stored in the section file will be used. If Fy is provided but Fu is omitted, Fu will be set to Fy.

ColdWork: Apply cold-work of forming. Default value is False.

Reserve: Apply inelastic reserve strength increase. Default value is False. If set to True, cold-work of forming will not be applied.

UseDSM: Use the Direct Strength Method. Default value is False.

Property **HasLicense**

Indicates if the instance of the Calculation class was successfully initialized with a license, and the license is still active – not released or dropped. Read only.

Sub **ReleaseLicense**()

To be called when done with the Calculation class so the license is immediately freed up. This is important for network license usage. If other instances of the Calculation class are still active, the license will not be released until all instances have released the license.

Calculation Methods

Function **SectionData**() As [SectionData](#)

Collects section information and interprets geometry for the current CFS section

Function **GrossProperties**() As [SectionProperties](#)

Calculates the gross section properties for the current CFS section

Function **NetProperties**() As SectionProperties

    Calculates the net section properties for the current CFS section

Function **EffectiveProperties**(*Forces* As SectionForces, *Spec* As Specification) As SectionProperties

    Calculates the effective section properties for the current CFS section

    Forces: Internal forces applied to the section

    Spec: Specification to use for the effective section calculation

Function **Strength**(*Spec* As Specification) As SectionStrength

    Calculates the fully braced strength for the current CFS section

    Spec: Specification to use for the strength calculation

Function **MemberCheck**(*Params* As MemberParams, *Forces* As SectionForces, [*FixCG* As Boolean]) As MemberUnityCheck

    Calculates a member design check for the current CFS section

    Params: Member parameters

    Forces: Internal forces applied to the section

    FixCG: Option to prevent axial eccentricities caused by a shift in the effective section centroid under axial compression. Default value is False.

Function **WebCheck**(*Params* As WebParams) As WebUnityCheck

    Calculates a web-crippling design check for the current CFS section

    Params: Member parameters

SectionData Class

Properties

ColdWork: Apply strength increase from cold-work of forming (Boolean)

ConnSpacing: Longitudinal spacing of shear connectors in built-up members (in)

Depth: Overall depth of section (in)

Description: Section description (read only)

E: Modulus of elasticity (ksi) (read only)

Flange: Flange width (in), negative return value indicates flange width varies

Fu: Section tensile strength (ksi), default is 50 ksi

Fy: Section yield strength (ksi), default is 50 ksi

HoleLength: Longitudinal dimension of holes in member (in) (read only)

HoleSpacing: Longitudinal center-to-center spacing of holes in member (in) (read only)

Lip: Lip length (in), negative return value indicates lip length varies

LipAngle: Angle of lip (radians), negative return value indicates lip angle varies

Material: Section material name (read only)

NumParts: Number of parts in the section (read only)

Project: Section project name (read only)

Radius: Inside bend radius (in), negative return value indicates radius varies

Reserve: Apply strength increase from inelastic reserve (Boolean)

RevDate: Section revision date (read only)

RevUser: Section revision user name (read only)

SectionType: Type of section (see SectionType options)

Thickness: Thickness of section parts (in), negative return value indicates thickness varies

UseDSM: Use the Direct Strength Method (Boolean, read only)

WebAngle: Angle of web (radians), negative return value indicates web angle varies

SectionProperties Class

ReadOnly Properties

Alpha: Angle of the major axis from the horizontal X axis, CCW positive (radians)

Area: Cross section area (in²)

Cw: Torsional warping constant (in^6)

Ic: Polar moment of inertia about centroid (in^4)

Io: Polar moment of inertia about shear center (in^4)

Ix: Moment of inertia about X axis (in^4)

Ixy: Product of inertia (in^4)

Iy: Moment of inertia about Y axis (in^4)

J: St. Venant torsion constant (in^4)

jx: Property used for lateral torsional buckling calculation (in)

jy: Property used for lateral torsional buckling calculation (in)

Rc: Polar radius of gyration about centroid (in)

Ro: Polar radius of gyration about shear center (in)

Rx: Radius if gyration about X axis (in)

Ry: Radius of gyration about Y axis (in)

Sxb: Section modulus about X axis for bottom fiber (in³)

Sxt: Section modulus about X axis for top fiber (in³)

Syl: Section modulus about Y axis for left fiber (in³)

Syr: Section modulus about Y axis for right fiber (in³)

Trace: Detailed calculation trace report (String)

Xl: Horizontal distance from centroid to left fiber (in)

Xo: Horizontal coordinate of shear center from centroid (in)

Xr: Horizontal distance from centroid to right fiber (in)

Yb: Vertical distance from centroid to bottom fiber (in)

Yo: Vertical coordinate of shear center from centroid (in)

Yt: Vertical distance from centroid to top fiber (in)

Zx: Plastic section modulus about X axis (in³)

Zy: Plastic section modulus about Y axis (in³)

SectionForces Class

Properties

B: Torsion bimoment (k-in²). For a C section with a vertical web, a positive bimoment causes compression in upper-left and lower-right quadrants of the section.

Dx: Horizontal deflection (in), use 0 to skip the deflection check and improve performance

Dy: Vertical deflection (in), use 0 to skip the deflection check and improve performance

Mx: Moment about X axis (k-in), compression on top is positive

My: Moment about Y axis (k-in), compression on right is positive

P: Axial force (k), compression is positive

Vx: Horizontal shear force (k)

Vy: Vertical shear force (k)

SectionStrength Class

ReadOnly Properties

Ae: Effective area at nominal compressive strength (in^2)

B: Torsion bimoment strength (k-in^2)

Flags: Combined flags indicating strength calculation notes (see NoteFlags)

Ixen: Effective moment of inertia about X axis at negative Mnx (in^4)

Ixep: Effective moment of inertia about X axis at positive Mnx (in^4)

Iyen: Effective moment of inertia about Y axis at negative Mny (in^4)

Iyep: Effective moment of inertia about Y axis at positive Mny (in^4)

Mxn: Negative moment strength about the X axis (k-in)

Mxp: Positive moment strength about the X axis (k-in)

Myn: Negative moment strength about the Y axis (k-in)

Myp: Positive moment strength about the Y axis (k-in)

P: Compressive strength (k)

Sxben: Effective section modulus for bottom fiber at negative Mnx (in^3)

Sxbep: Effective section modulus for bottom fiber at positive Mnx (in^3)

Sxten: Effective section modulus for top fiber at negative Mnx (in^3)

Sxtep: Effective section modulus for top fiber at positive Mnx (in^3)

Sylen: Effective section modulus for left fiber at negative Mny (in^3)

Sylep: Effective section modulus for left fiber at positive Mny (in^3)

Syren: Effective section modulus for right fiber at negative Mny (in^3)

Syrep: Effective section modulus for right fiber at positive Mny (in^3)

T: Tension strength (k)

Trace: Detailed calculation trace report (String)

Vx: Horizontal shear strength (k)

Vy: Vertical shear strength (k)

MemberParams Class

Properties

BracedFlange: Braced flange (see Flange options)

BucklingTheory: Use elastic theory for global buckling calculations (Boolean)

Cbx: Coefficient for bending about X axis, default is 1

Cby: Coefficient for bending about Y axis, default is 1

Cmx: Coefficient for moment about X axis, default is 1

Cmy: Coefficient for moment about Y axis, default is 1

DxMax: Maximum allowable horizontal deflection (in)

DyMax: Maximum allowable vertical deflection (in)

ex: Axial load eccentricity in the X direction (in)

ey: Axial load eccentricity in the Y direction (in)

Kf: Flange rotation stiffness (k-in/rad/in)

Kt: Effective length factor for twisting

Kx: Effective length factor for buckling about X axis

Ky: Effective length factor for buckling about Y axis

Lm: Unbraced length between distortional buckling restraints (in)

Lt: Unbraced length for twisting (in)

Lx: Unbraced length for buckling about X axis (in)

Ly: Unbraced length for buckling about Y axis (in)

MxMax: Maximum moment about X axis (k-in), used to adjust vertical deflection

MyMax: Maximum moment about Y axis (k-in), used to adjust horizontal deflection

Pdelta: Indicates whether applied moments include P-δ effects, default is False

R: Moment reduction factor for fully braced tension flange

Spec: Specification to use for the member check (see Specification options)

MemberUnityCheck Class

ReadOnly Properties

B: Torsion bimoment strength (k-in²)

BMxMy: Unity check for combined torsion/bending

Dx: Unity check for horizontal deflection

Dy: Unity check for vertical deflection

Flags: Combined flags indicating strength calculation notes (see NoteFlags)

Mx: Moment strength about X axis (k-in)

MxVy: Unity check for combined shear/bending about X axis

My: Moment strength about Y axis (k-in)

MyVx: Unity check for combined shear/bending about Y axis

P: Axial strength (k)

PMxMy1: Unity check #1 for combined axial/bending

PMxMy2: Unity check #2 for combined axial/bending

Trace: Detailed calculation trace report (String)

Vx: Shear strength in horizontal direction (k)

Vy: Shear strength in vertical direction (k)

WebParams Class

Properties

Dir: Load direction (1=vertical, 2=horizontal)

Fastened: Bearing flange is fastened to the support (Boolean)

Lend: Distance from edge of load to end of member (in)

Lload: Distance from edge of load to edge of opposite load (in), use large value if no opposing concentrated load

M: Moment at the point of bearing (k-in)

N: Flange bearing length (in)

P: Concentrated load or reaction (k), positive load is on bottom flange (or left flange for horizontal load)

Spec: Specification to use for the web check (see Specification options)

WebUnityCheck Class

ReadOnly Properties

Flags: Combined flags indicating strength calculation notes (see NoteFlags)

M: Unity check for bending only

P: Unity check for web-crippling only

PM: Unity check for combined bending/web-crippling

Trace: Detailed calculation trace report (String)

SectionType Enum

Constants

Unknown = 0 (returned if loaded section does not match one of the other profiles)
Cee = 1
CeeStiffened = 2
Zee = 3
ZeeStiffened = 4
DoubleCee = 5
DoubleCeeStiffened = 6
Box = 7
Tube = 8
Angle = 9
AngleStiffened = 10
Hat = 11
Cylinder = 12

Specification Enum

Constants

AISI1999USASD = 0
AISI1999USLRFD = 1
AISI2001USASD = 2
AISI2001USLRFD = 3
AISI2001MexASD = 4
AISI2001MexLRFD = 5
AISI2001CanLSD = 6
AISI2004USASD = 7
AISI2004USLRFD = 8
AISI2004MexASD = 9
AISI2004MexLRFD = 10
AISI2004CanLSD = 11
AISI2007USASD = 12
AISI2007USLRFD = 13
AISI2007MexASD = 14
AISI2007MexLRFD = 15
AISI2007CanLSD = 16

AISI2010USASD = 17
AISI2010USLRFD = 18
AISI2010MexASD = 19
AISI2010MexLRFD = 20
AISI2010CanLSD = 21
AISI2012USASD = 22
AISI2012USLRFD = 23
AISI2012MexASD = 24
AISI2012MexLRFD = 25
AISI2012CanLSD = 26
AISI2016USASD = 27
AISI2016USLRFD = 28
AISI2016MexASD = 29
AISI2016MexLRFD = 30
AISI2016CanLSD = 31
AISI2018USASD = 32
AISI2018USLRFD = 33
AISI2018MexASD = 34
AISI2018MexLRFD = 35
AISI2018CanLSD = 36

Flange Enum

Constants

None = 0
Bottom = 1
Top = 2
Left = 3
Right = 4

CalculationException

Messages

No license found
License was released
License check failed
Error loading section: …
Geometry error: …
Properties error: …
No section has been loaded or defined yet
Section has no DSM data
Invalid section definition
Invalid yield stress
Invalid tensile strength
Invalid thickness
Invalid radius
Invalid section depth
Invalid flange width

Invalid lip length
Invalid web angle
Invalid lip angle
Invalid section type
Invalid connector spacing
Invalid hole length
Invalid hole spacing
Invalid specification
Invalid unbraced length
Invalid effective length factor
Invalid braced flange
Invalid moment reduction factor
Cannot apply one-flange bracing to a fully braced member
Invalid value for Cb
Invalid value for Cm
Invalid value for maximum deflection
Invalid load direction
Invalid bearing length

NoteFlags Enum

Constants (additive)

ThinTubeWall = 1: D/t exceeds maximum for cylindrical tube
SlenderElement = 2: w/t exceeds maximum for an element
LongEdgeStiffener = 4: D/w or do/bo exceeds maximum for an edge stiffener
EdgeStiffenerAngle = 8: Edge stiffener angle outside limits
SlenderColumn = 16: KL/r exceeds maximum for a column member
SheathedColumnLimits = 32: Section does not meet all req's for sheathed column provisions
SheathedColumnFastener = 64: Sheathed column fastener assumed to be at middle of flange
ConnectorSpacing = 128: a/ri exceeds 0.5KL/r
BendRadius = 256: R/t exceeds limit
HoleDepthRatio = 512: dh/h exceeds limit
HoleDepth = 1024: Hole depth exceeds limit
HoleLength = 2048: Hole length exceeds limit
HoleSpacing = 4096: Clear distance between holes less than limit
YieldStress = 8192: Yield stress exceeds Specification limit
TorsionCheck = 16384: Unable to check torsion with overridden Cw
Stiffeners = 32768: Number of intermediate stiffeners exceeds limit

Appendix 1 – Changes from CFS12

The CFS13 module has implemented the following interface changes over the CFS12 module:

SectionData Class: HoleLength and HoleSpacing are now read-only.

SectionProperties Class: Added Zx and Zy plastic section modulus properties.

DefineSection Method: SectionData.Depth is not required for angle sections.

Multiple simultaneous instances of the Calculation object are fully supported for multi-threaded applications.