Release PCB-Investigator 9.0 Automation Interface update

New features:

IAction

Remove of
ID_DRAG_N_DROP_FOR_CMPS = 128

Add new option:
Cut negative objects out of underlying objects on all layers.
ID_REMOVE_NEGATIVE_OBJECTS = 178

IAutomation

Suppress F5 Button to update menu and dialogs.
static bool SuppressStandardKey
Ask Option for plugin directory of user. This is used to load private plugins by starting PCB-Investigator.
static string DeveloperPlugInDirecotry
Unzip files with temporary attribute to hold them in RAM instead of flushing to disk.
static bool UnzipWithTempFileAttribute
Use DEBUG Output add more information to error log and gives developer more details and warnings by using interface methods.
static bool DEBUG_Output
Check PCBI allows using API interface. It is possible to get an API license without option to show PCB-Investigator and maybe you can show PCB-Investigator but have no license to add your own API methods.
static bool CheckPCBILicenceAPI()
End date of your PCB-Investigator license. It is important to init the options before, this will be done e.g. by create a IPCBIWindow.
static DateTime GetEndDateOfLicence()
Set User Name for SOD License System.
static bool SetSODLicence(string UserName, string Password)
Set Floating Server IP with Port.
static bool SetFloatingLicence(string IP, string Port)
Load a license file with plugins and PCB-Investigator licenses.
static bool LoadLicenseFile(string FullPathOfLicenseFile)
Ask for installation type.
static PCBIINSTALLATIONTYPE InstallationType

Serialize a file with PCBI standard settings.
static bool XMLSerializeToFile(Plugin.Interfaces.IFileData xmlFile, object serializableObject)

Deserialize a xml file with PCBI standard settings.
static bool XMLDeserializeFromFile<T>(Plugin.Interfaces.IFileData xmlFile, ref T deserializedbject)

new Enum:
public enum PCBIINSTALLATIONTYPE

ICMPObject

Returns the transformation matrix with translation, rotation and mirroring
MathUtils.MatrixD GetTransformMatrix(out bool isMirrored)

IDrawingParameters

IDrawingParameters:

Background color
Color BackColor
If true it ignores the specific color of the component.
bool IgnoreObjectColor = false;

IEdge

Split this arc in smaller elements depending on the ArcLength parameter.
List<IArcEdge> SplitArc(double ArcLength)

New Enums:
WarningCode
ErrorCode
MatrixStackupCheckResult

IFilter

Get sub details of special pad, this can contain all types of IODBObjects.
static List<IODBObject> GetToolDefinitionSpecialSubElements(IPCBIWindow PCBIWindow, string CADConformNameOfSpecialSymbol)
Adds a octagon tool definition
static int AddToolDefinitionOct(Automation.IODBLayer Layer, double diameter, double octagonHeight, double octagonWidth, double corners, int ToolNr = 0)
Add a definition of oval symbol.
static int AddToolDefinitionOval(Automation.IODBLayer Layer, double diameter, double ovalHeight, double ovalWidth)

IMath

Combine both rectangles.
    static RectangleD Union(RectangleD a, RectangleD b, bool IgnorEmptyRectangle)
Calculate the length of the edge element.
    static double GetLength(IEdge e)
Calculate the distance between a point and a polygon.
    static double DistancePointToPolygon(PointD P, IPolyClass Polygon, out PointD PonPolygon)
Distance Point to arc calculates the distance from a point P to the arc edge Arc.
    static double DistancePointToArc(PointD P, IArcEdge Arc, out PointD PonArc)
Cosinus with special cases for rounding problems.
    static double Cos(double angleGRAD)
Sinus with special cases for rounding problems.
    static double Sin(double angleGRAD)

IMatrix

Rename
GetRawIndexByName to GetRowIndexByName
DelateLayer to DeleteLayer

List of all board layer names contain all layers with context board.
    List<string> GetAllBoardLayerNames(bool ToLower)
List of all signal layer names.
    List<string> GetAllSignalLayerNames(bool ToLower)
Check whether copper layer exists, prepregs are between copper layers and all copperlayers/prepregs have a thickness >0.
    MatrixStackupCheckResult CheckStackup(IStep Step)

IObject

Resets the set color.
    virtual void ResetColor()

IODBLayer

Rotate the layer by the specified angle (degree).

bool RotateLayer(double Angle)

Removes all objects on this layer (no Undo)

void RemoveAllObjects()

Flood layer with surfaces, this use pcb outline for edge.

bool FloodLayer(IPCBIWindow PCBIWindow, double FloodDistance = 5, double DistanceObjects = 5, double MinSizeFlood = 4, double MinAreaFlood = 10, double DistanceDrills = -1)

Oversize or undersize the layer, with option for only selected and values in percentage.

bool Oversize(double Oversize, bool OnlySelected, bool IsPercentage = false, double MaxOversize = 1000000)

IODBObject

Rotate the object.

void Rotate(double Angle, bool AddToUndoList = true)

IPictureLayer

Get the original image used by this picture layer

This method use a clone to give you no reference to the internal image, the image is a shadow copy if you dispose the original image before you use the new image it will be cleared too.

Bitmap GetLayerImage()

IPCBIWindow

Load tgz data from stream e.g. use MemoryStream.
async System.Threading.Tasks.Task<bool> LoadDataFromTGZStreamAsync(Stream DataStream)
Load data async from all available formats (depending on your license).
async System.Threading.Tasks.Task<KeyValuePair<PCBI.Automation.IPCBIWindow.LoadInformation, PCBI.ImportOptions.FormatTypes>> LoadDataAsync2(string FullPath)
Save ODB++ data in zip stream. The memory stream can be corrupted if a error occurs (Check error log for this case).
async System.Threading.Tasks.Task<bool> SaveDataInZipStreamAsync(MemoryStream ms)
Delegate for closing request from e.g. IPluginOpenFileControl
delegate void CloseRequestEvent(DialogResult result, Plugin.Interfaces.IPlugin caller);
Load rules for PlugIn (Each plugin have his own container). Analsysis scripts are alle manged over scripting, if you change the Name from PCB-Investigator-Scripting to other name it do not find the analysis script settings again.
PCB_Investigator.Automation.Rules.RulesContainer GetRuleSetting(Plugin.Interfaces.IPCBIAnalysisPlugIn)
Set rules for one plugin with the RulesContainer.
bool SetRuleSetting(PCB_Investigator.Automation.Rules.RulesContainer newRuleSet)
Save rule settings for all analysis plugins.
bool SaveRuleSettings()
Load all settings for all available analysis plugins.
PCB_Investigator.Automation.Rules.DesignRulesSet GetCurrentRuleSets()
Ask all plugins for there standard rules and combine them in one rulesSet.
PCB_Investigator.Automation.Rules.DesignRulesSet GetStandardRuleSets()

Import setting from xml file.
PCB_Investigator.Automation.Rules.DesignRulesSet ImportRuleSets(Plugin.Interfaces.IFileData FileData)
Preview image is created from outline with drills. Simple without special pads and only one drill layer.
The image will be created of the loaded job, if no job is loaded the return bitmap is black.
It will use the current step for pcb outline and drills.
Bitmap CreatePreviewImage(Size ImageSize, bool ShowComponents)

IPin

Set special text to pin, this must be activated in component view setup or IPCBIWindow.ActivatePinSpecialInfo().
string GetPinSpecialInfo(ICMPObject IcmpObject)
bool SetPinSpecialInfo(ICMPObject IcmpObject, string value)

IPolygon

Add an edge (ILineEdge or IArcEdge) to the polygon.
Add an arc Edge to the polygon (its create the IArcEdge internal)
Add an ILineEdge to the polygon.
    void AddEdge ()
Transforms the poylgon by the given Matrix
    void Transform(PCBI.MathUtils.MatrixD m)
Check the whether polygon is closed?
    bool IsClosed()
Count of edges in polygon.
    int EdgeCount

A special Debug View Class is added to have Image and some special information while debugging IPoly-Class elemenets.

ITextSpecificsD

Set Width of text in mils. This means the diameter of each line in each letter has the fix value.
    void SetWidthText(double widthInMil)

ISurfaceSpecificsD

Remove small line and arc elements
    bool CleanUp(double errorLevel)
Move and rotate the surface.
    void Transform(PointD Offset, double Angle)

IPackageSpecificsD

Get image of package width specific size and colors.
    Bitmap GetPackageImage(int width, int height, Color bkgColor, Color bodyColor, Color bodyFillColor, Color textColor, Color pinColor, double InflateSize = 10)
Mirror package in X direction.
    void MirrorX()
Mirror package in Y direction.
    void MirrorY()

IStep

Set the PCB contour from an list with single Poly Class objects.
void SetPCBOutline(MathUtils.IPolyClass PCBOutline, bool AddUndo = false)
Creates the outline from a list of objects.
bool SetPCBOutline(List<IODBObject> objects, bool useMiddleLine, double minLineLengthMils = 1.0)
Create DXF File for Layerlist with some parameters and colors are used from layer (only some DXF colors supported e.g. red, grey, yellow...).
void SaveDXFFile(string FullPath, List<ILayer> LayerForOutput, bool Skeleton, bool CombineNets, bool onlySelected, bool connectHolesToIles, double flatten, bool onlyPolygons, bool addProfilContour, bool WriteInfo)
Set the steps to repeate in this step, all child steps must have a existing name to repeat them.
void SetChildSteps(List<StepAndRepeatClass> ChildSteps, bool UpdateControls)
Add single step as child to current step.
bool AddChildStep(StepAndRepeatClass ChildStep)
Set the datum fot step header. This is used for Step and Repeat in other steps where this is a child step.
void SetDatumStepHDR(PCBI.MathUtils.PointD Datum)
Set the origin of step header. This is used for Step and Repeat in other steps where this is a child step.
void SetOriginStepHDR(PCBI.MathUtils.PointD Origin)
Reads Gerber or Excellon data, errors are reported in the ErrorLog.
string AddGerberLayer(string FullPath, bool OverwriteIfExists, FormatTypes Type, int Leading-Numbers = 0, int TrailingNumbers = 0, bool? unitInch = null, bool TryAutoRecognition = true, bool AddUndo = false, omit_zeros_t LeadingOrTrailingZeros = omit_zeros_t.EXPLICIT, bool ImproveAreaFills = true, bool AxisSelectionABandXYChange = false)
Select all objects of the choosen net.
void SelectNetElements(string NetName, bool CaseSensitive)
Search for the right number to the NetName.
int GetNetNrFromNetName(string NetName, bool CaseSensitive)
Check the netlist for count of nets in this step (no child steps included!).
int GetNetCount()
Search for an net with NetName.
INet GetNet(string NetName, bool CaseSensitive)

Fills the MemoryStream with rgb image data (3 bytes/pixel). Layer-Color is used.
Components are drawn with a filled body, without pins and without any label
A seperate licence is necessary to use AOI!
bool AOIHighResolutionMemoryExport(ref MemoryStream ms, ICMPLayer Layer, MathUtils.RectangleD ClippingRectangle, int DPI, bool drawOnlySelected, Color BackColor, out int widthPx, out int heightPx, PCBI.Automation.IPCBIWindow.ProgressChanged onProgressChanged = null)

The step will be flatted, all repeated objects from other steps are placed on the new step.
bool FlattenStep(string newStepName, bool saveRAM, bool BoardIDAppendForCMPRef, char Separator = '_')
The step will be flatted, all repeated objects from other steps are placed on the new step. But only for the layers in LayerNamesToFlatten list (use names toLower).
bool FlattenStepOnlySpecificLayers(string newStepName, bool OnlyObjectsInProfile, List<string> LayerNamesToFlatten)

PCBI resources

New PCBIWindows:

PCBIEnerTextDialog
PCBIOpenFileDlg
PCBIResultDialog
PCBISaveFileDlg