

Files Used by SIMION 7.0

Introduction

SIMION creates and uses a collection of files. Each type of file generally has its own unique extension (e.g. **.PA**) to signify to you and SIMION what it is. Files are used to store electrode geometry, potential arrays, contour intervals, trajectory parameters, and more.

It is important that you observe SIMION's file conventions. The following is a list of the files SIMION uses:

Files Found in Project Directories

The following is a list of files that may appear in a project's directory with SIMION 7.0 versus 6.0 file compatibility information:

File: *.ANN (All files with .ANN extensions)

Created by:	SIMION in the PRINT OPTIONS SCREEN
Used by:	SIMION in the PRINT OPTIONS SCREEN
Function:	A Collection of Saved Annotations
7.0 – 6.0 Compatibility:	New Feature of 7.0

SIMION creates *.ANN files via the **Save** button on the Print Options Screen. These files contain the annotations in use at the time of the save. Annotations can be reloaded via the **Load** button on the Print Options Screen. *See Appendix G for more information.*

File: *.CON (All files with .CON extensions)

Created by:	SIMION in the CONTOUR MODE
Used by:	SIMION in the CONTOUR MODE
Function:	A Collection of Contour Intervals
7.0 – 6.0 Compatibility:	Same File Format

SIMION creates *.CON files via the **Save** button on the Contour Mode Screen. These files contain a list of potential and gradient contour intervals for later use via the contour file **Load** button.

File: *.FLY (All files with .FLY extensions)

Created by:	SIMION in the TRAJECTORY DEFINITIONS
Used by:	SIMION in the TRAJECTORY DEFINITIONS
Function:	Parameters for Grouped Ion Definitions
7.0 – 6.0 Compatibility:	Same File Format

When you enter the parameters for a specific collection of grouped ion definitions SIMION allows you to save these parameters as a *.FLY file. This allows you to later recall the grouped ion definitions without having to remember and re-enter them.

Files Used by SIMION 7.0

File: *.GEM (All files with .GEM extensions)

Created by: YOU with Editor (e.g. EDY) as ASCII file
Used by: SIMION for Defining Electrode/Pole Geometry
Function: Electrode/Pole Geometry Definition Files
7.0 – 6.0 Compatibility: Same File Format (*some new 7.0 commands*)

Geometry files are used as an alternate way to define complex 2D and 3D electrode/pole geometry. This is an advanced SIMION feature. *See Appendix J for more information.*

File: *.JOB (All files with .JOB extensions)

Created by: SIMION in the VIEW OPTION
Used by: SIMION in the VIEW OPTION
Function: Ion Optics Bench Definition File
7.0 – 6.0 Compatibility: Upward and Downward Compatible

When you define an ion optics workbench of instances the **Save** button on the Normal Mode Screen (*in View*) allows you to save this definition as an *.JOB file. This allows you to later load the *.JOB file and have SIMION automatically establish the workbench for you (*including loading potential arrays and restoring potentials*).

The *.JOB file formats differ from 7.0 to 6.0. However, they are fully upward compatible from 6.0. SIMION 6.0 can read 7.0 *.JOB files but the long filename information will be lost if saved by 6.0. It is recommended that if an *.JOB file is to be read and saved by both 6.0 and 7.0 that conventional DOS 8.3 file naming conventions be used.

File: *.ION (All files with .ION extensions)

Created by: SIMION in the TRAJECTORY OPTION
Used by: SIMION in the TRAJECTORY OPTION
Function: Parameters for Individual Ion Definitions
7.0 – 6.0 Compatibility: Upward but NOT Downward Compatible

When you enter the parameters for a specific collection of individual ion definitions SIMION allows you to save these parameters as an ASCII *.ION file. This allows you to later recall the individual ion definitions without having to remember and re-enter them. These ASCII files can be created externally by you with an editor or another program.

The *.ION file format has been expanded in 7.0 to include comments and units (*e.g. whether ion's parameters are in workbench mm or some instance's gu*). The actual formats used by 7.0 and 6.0 *.ION files are described at the end of this appendix.

File: *.KEPT_TRAJ (All files with .KEPT_TRAJ extensions)

Created by: SIMION in the SAVE IOB FILE OPTION
Used by: SIMION in the LOAD IOB FILE OPTION
Function: Holds Kept Ion Trajectories for IOB Auto-Load
7.0 – 6.0 Compatibility: NOT Compatible (*new 7.0 feature*)

When you save an *.JOB file and you have kept ion trajectories active (*via the Keep button*) SIMION 7.0 will ask you if you want the kept ion trajectories saved as an auto-loading file. If you click **Yes** a kept trajectory file with the .JOB file's name and the .KEPT_TRAJ extension will be saved in the *current* project directory.

File: *.PA? (All files with .PA? extensions)

Created by:	SIMION
Used by:	SIMION
Function:	Holds an entire potential array
7.0 – 6.0 Compatibility:	Same File Format (<i>new file type – see below</i>)

A **.PA?** file contains all the data for an entire potential array in machine binary image storage format. These files preserve potential arrays between SIMION sessions. The ? signifies any legal file naming character (e.g. **.PA**, **.PA1**, **.PAA**). The format used by ***.PA** and ***.PA#** files is described at the end of this appendix to allow reading and creation via your own C programs.

A new file type with the extension ***.PA_** may now be automatically created by Refine in SIMION 7.0. This is a fast scaling file for all non-zero electrode/pole points that are not defined as adjustable electrode points (e.g. integer potential values from 1-30). *See Chapter 6 for more information.*

File: *.PRG

Created by:	YOU with Editor (e.g. EDY) as ASCII file
Used by:	SIMION for User Programs
Function:	Holds User Programs
7.0 – 6.0 Compatibility:	Same File Format (<i>new 7.0 commands</i>)

These files contain user programs generated by the user. For example: The file **TEST.PRG** is the user program file for the potential array **TEST.PA** (or **TEST.PA0**). User programming is an advanced SIMION feature. *See Appendix I for more Information.*

File: *.REC

Created by:	SIMION in DATA RECORDING
Used by:	SIMION in DATA RECORDING
Function:	Holds Data Recording Definitions
7.0 – 6.0 Compatibility:	Same File Format

These files contain data recording definitions. The data recording feature of SIMION allows you to save and load data recording definitions as ***.REC** files.

Temporary Ion Trajectory Files

SIMION 6.0 stored ion trajectories in a temporary file called “**IONTRAJ.DRW**” in the currently active (*project*) directory.

SIMION 7.0 stores ion trajectories in two unique temporary files (*the extra file is for kept trajectories*) in the currently active (*project*) directory. These files have the name format: “**trj????.tmp**” where **????** is a hexadecimal unique number. This approach allows more than one running version of SIMION 7.0 to use the same project directory simultaneously.

Files Used by SIMION 7.0

Important Files in the C:\FILES.GUI Sub Directory

The following GUI and general SIMION support files are found in the C:\FILES.GUI sub directory. It is useful to know about them.

SIMION 7.0 (a Win32 GUI version) has been designed not to file conflict with SIMION 6.0 (a DOS GUI version). Thus it uses different names and contents for certain files (discussed where appropriate below).

File: GUI?????.TMP (Same File Naming format for 7.0 and 6.0)
Created by: GUI
Used by: GUI
Function: GUI's temporary GUI Error File.

The GUI?????.TMP file retains status and error information for a currently running GUI application (this is done to avoid sharing violations when multiple GUI applications are concurrently running). Automatically managed by GUI based programs. *Program lockups and rude behavior on your part can create a collection of .TMP files. Feel free to erase them to clean out the clutter.*

File: ERRORS.GUI (Same File Name in 7.0 and 6.0)
Created by: GUI
Used by: GUI
Function: GUI's Error File.

The ERRORS.GUI file retains status and error information for a GUI application. *If something strange is happening with lots of beeping, quit SIMION and take a look at this file.* Automatically managed by GUI based programs.

File: WINI.GUI (File Name used by SIMION 7.0)
File: INI.GUI (File Name Used by SIMION 6.0)
Created by: GUI
Used by: GUI
Function: GUI's Personality file.

The WINI.GUI file retains the color palette, delays, sounds, mouse speed, and other appearance items. Automatically managed by Win32 GUI based programs. *If you erase this file, the Win32 GUI will assume its default Window personality.*

File: WPLOTTERS.GUI and WANNOTATE.GUI (File Names used by SIMION 7.0)
File: PLOTTERS.GUI and ANNOTATE.GUI (File Names Used by SIMION 6.0)
Created by: GUI
Used by: GUI
Function: GUI's Print Personality & Annotations files.

The WPLOTTERS.GUI file retains the current Windows printing options. Automatically managed by Win32 GUI based programs. *If you erase WPLOTTERS.GUI, the Win32 GUI will assume its*

Files Used by SIMION 7.0

default Windows printing personality. The WANNOTATE.GUI file retains the last print annotations defined in a Win32 GUI based program session.

File: WVIDEO.GUI (SIMION 7.0 – Windows Video Information)
File: VESA.GUI (SIMION 6.0 – DOS Video Information)
Created by: GUI
Used by: GUI
Function: GUI's Video Personality file.

The WVIDEO.GUI file retains the current Windows video options. Automatically managed by Win32 GUI based programs. *If you erase this file, the GUI will assume its default Windows video options.*

File: ?_WDIR.GUI (File Names Used by SIMION 7.0 – Long Names)
?_WSIZE.GUI
File: ?_DIR.GUI (File Name Used by SIMION 6.0 – Short Names)
Created by: GUI
Used by: GUI
Function: GUI's Directory Scan for a Drive.

The ?_WDIR.GUI file retains the last *long* name directory scan for drive ? (e.g. C_WDIR.GUI for C drive). The ?_WSIZE.GUI file retains the directory size scan for drive ? (e.g. C_WSIZE.GUI for C drive). Automatically managed by the Win32 GUI File Manager (recreated with Scan button). *If you erase either of these files, the Win32 GUI File Manager will automatically rescan the drive.*

File: INITIAL.TBL (Created Random Number Generator Table)
File: ACTIVE.TBL (Active Random Number Generator Table -- user optioned file)
Created by: INITIAL.TBL – SIMION at Program Startup
ACTIVE.TBL – User Copied INITIAL.TBL image
Used by: SIMION's Random Number Generator
Function: To Define Pseudo-Random Number Sequences

These files support random number generation for user programs (Appendix D). Their function and use options are discussed in Appendix E (E-15).

The Format Used with SIMION 7.0 .PA Files (same as 6.0)

The following is provided for those who may want to generate or read SIMION 7.0 .PA or .PA# files. *The definitions below assume C conventions:*

```
#define PLANAR          1          /* planar symmetry */
#define CYLINDRICAL     0          /* cylindrical symmetry */

#define MIRROR_X        1          /* array mirrored in x */
#define MIRROR_Y        2          /* array mirrored in y */
#define MIRROR_Z        4          /* array mirrored in z */
#define MAGNETIC_PA     8          /* is magnetic potential array
else assumed electrostatic array*/
```

Files Used by SIMION 7.0

```
int ng = 100; /* number of grid points between poles */

struct header_3d /* header image for SIMION 7.0 PA files */
{
    long mode; /* mode must be -1 */
    long symmetry; /* PLANAR or CYLINDRICAL */
    double max_voltage; /* max voltage allowed for pa */
    long nx; /* array's x dimension size */
    long ny; /* array's y dimension size */
    long nz; /* array's z dimension size */
    long mirror; /* (MIRROR_X | MIRROR_Y |
MIRROR_Z | MAGNETIC_PA) +
(ng << 4) */
};
```

Note: mirror contains mirroring flags, magnetic PA flag, and shifted copy of ng.

File Format:

```
struct header_3d /* byte packed - no skipping */
double points [ nz ] [ ny ] [ nx ]; /* array in internal binary format */
```

The values of the points are as follows:

```
Non-electrode points = Potential;
Electrode points = Potential + 2.0 * max_voltage
```

Note: Max_voltage should be selected to be well above the maximum positive potential (e.g. 10,000 volts). SIMION manages max_voltage and will automatically increase it as the need arises during array usage.

If you are creating .PA# files, the Potential values for all points of electrode or pole number one must be 1.0. Likewise points for the second adjustable electrode/pole must be 2.0 and so on.

Trick: If you have a previously defined potential field (*be it an analytical expression, experimental data, from another program or whatever*) that can be used to estimate the potentials at array point locations (*by some method*), a potential array can be created from this information. Remember: There is **NO** need to include electrode or pole points as the potential array should **NOT** be refined.

ASCII Format Used in Individual Ion Definition Files .ION

The ASCII file format has one (*and only one*) ion definition per line with parameters in the order below separated by commas. Parameters can be skipped (*using commas*) or omitted (*trailing parameters*). **SIMION will automatically assume default values for any skipped/omitted parameters.**

```
TOB, MASS, CHARGE, X, Y, Z, AZ, EL, KE, CWF, COLOR
0, 100, 1, 0, 0, 0, 0, 0, 0.1, 1, 0(black) Default Values
```

The SIMION 7.0 .ION file format has been extended to include a unit definition number (e.g. *instance number*) **and** comment lines. To be considered a comment line the **first** non-space character in the line must be a semi-colon (;). The **optional** unit definition number must be the first line in the .ION file (*if supplied*) as a comment containing **only** the instance number as shown below:

Files Used by SIMION 7.0

; 1 (ion definitions are in instance number 1 units)
0,100,1 (first ion's definition)
; A comment line (comment line is ignored)

Allowed values for the unit definition number are *either* an instance number $1 \dots n$ (where n = the highest instance number) *or* **0** which means workbench coordinates.

Note: SIMION 7.0 always saves an .JON file with a units definition line as the first line. This makes 7.0 .JON files unreadable by SIMION 6.0 unless you use an editor to remove the first comment line.

THIS PAGE INTENTIONALLY LEFT BLANK