Tester Data Text Format

Introduction:

Tester Data Text Format was defined by Spry Software, Inc and is intended to be a generally usable, user-friendly format for exchange of information about testing of integrated circuits. It is not necessarily intended as a replacement for STDF, but rather as a more convenient format for exchange of data between companies, users or tools. Its implicit csv format, clear linking of parts to test results and provision of fields such as test stage and test map that match contemporary manufacturing practices all make it more user friendly than its binary predecessor.

Spry Software reserves the exclusive right to change the format, but welcomes input on how to make it more useful. If you have suggestions for improvement please communicate them to tdtf@sprysoftware.net.

Organization:

TDTF is organized into a series of sections, some required and some optional. Each is introduced by a line with the word “Section” followed by a comma and a section name. Some sections have a required header and some can appear multiple times, as described in the section header below.

All rows must be composed of fields separated by commas. Values may be surrounded by quotes; this is necessary if a value contain commas.

Typical section order:

Format

Lot

Test Insertion

Equipment

Bins (optional)

Tests (optional, required if parametric data is present in PartResults)

Wafer (optional)

PartResults (optional)

If the file includes multiple wafers (this often happens for WAT data), the Wafer and PartResults sections can repeat. An example TDTF without wafers is at https://sprysoftware.net/tdtf/short_ft4.TDTF . An example file with wafer sort data is available at https://sprysoftware.net/tdtf/short_ws2.TDTF .

Format Section:

Summary: Intended to indicate the version of TDTF used in this file

Keys:

FormatVersion: decimal version number

Lot Section:

Summary: Intended to contain information descriptive of the lot as a whole

Format: 2 fields per line, key and value, no header row

Keys:

Lot: the lot identifier, without extraneous characters that should be in other fields. Normally this is the MES lot identifier

WaferLot: the original fab or foundry lot from which this lot was made. Can be the same as Lot.

ProductionStatus: Typically P for production, E for engineering, users can define others

PartFamily: user defined

Test Insertion:

Summary: Intended to contain general information about this testing

Format: 2 fields per line, key and value, no header row

Keys:

TestType: Typically WAT, WS or FT. WS should include any probing of individual die still on a wafer.

TestStage: Typically WAT, CP1, CP2, CP3, FT1, etc, but users can define different values if necessary. Often but not always equivalent to MIR.test_cod

TestStep: This relates to the step within the particular stage, usually with relation to the company's re-test process and nomenclature

Program: The name of the test program. Equivalent to MIR.job_nam

ProgramRevision: The revision name or number of the test program. Equivalent to MIR.job_rev

SetupTime: Dates should be in the format yyyy-MM-dd HH24:mm:ss, for example 2020-11-05 13:18:21, preferably in UTC. This field is equivalent to MIR.setup_t

StartTime: same format as SetupTime, equivalent to MIR.start_t

EndTime: same format as SetupTime, equivalent to MRR.finish_t

Operator: equivalent to MIR.oper_nam

UserText: equivalent to MIR.user_txt

DataLogFile: the name of a source file from which this TDTF was created, for example an STDF file

Temperature: equivalent to MIR.tst_temp

Equipment:

Summary: Intended to store all information related to equipment used in the process described in the current file

Format: 3 fields per line, key, equipment type or brand, name of the specific piece of equipment

Tester: equivalent to MIR.tstr_typ and MIR.node_nam

ProbeCard: equivalent to SDR.card_typ and SDR.card_id

Handler: equivalent to SDR.hand_typ and SDR.hand_id

LoadBoard: equivalent to SDR.load_typ and SDR.load_id

Dib: equivalent to SDR.dib_typ and SDR.dib_id

Cable: equivalent to SDR.cabl_typ and SDR.cabl_id

Contactor: equivalent to SDR.cont_typ and SDR.cont_id

Laser: equivalent to SDR.lasr_typ and SDR.lasr_id

Bins:

Summary: contains summaries for the whole file and each site of the count for each hard and soft bin. Can be present even if no part results are present and can be omitted completely from file (for example, for WAT data).

Format: has initial format line listing the required fields and then the site-specific columns. Here is an example header + initial data line:

```
BinType,Number,Name,PF,relatedHardBin,TotalCount,Site1Count,Site2Count,Site3Count,Site4Count
```

```
HARD,"1","Good",P,,810,195,190,204,221
```

Columns:

BinType: HARD or SOFT

Number: numeric representation of bin

Name: optional text description of bin. Column should be present even if no values are defined

PF: P for pass, F for fail, empty for unknown

RelatedHardBin: empty for hard bins, Number value for related hard bin for soft bin rows

TotalCount: total number of units for this bin in the file

SiteXCount: number of bins in site X in the file, where X is a numeric value.

Tests:

Summary: contains one row per test represented in the file. This section is required if the file has a PartResults section, but there should only be one Tests section even if there are multiple PartResults sections.

Format: contains a header row for ease of reading, even though fields are currently fixed. Example:

Number,Name,Type,LowLimit,HighLimit,ResultScale,LowLimitScale,HighLimitScale,Unit

101,"Cont_IN",Parametric,-1.0,0.2,0,0,0,VOLTS

All of these fields have equivalents in STDF.

Wafer:

Summary: contains identifying information about one wafer. This section can be repeated, usually with a paired PartResults section for each wafer.

Format: contains key and value columns:

Keys:

WaferId: can be either the 1 or 2 digit numeric wafer id, or the whole id from WIR.wafer_id

WaferText: if WaferId is numeric and a longer text exists, for example from the value scribed on the wafer, it can be stored in WaferText.

Part Results:

Summary: contains identifying information and test results for each die, part or test site (collectively, part) on a wafer or in a lot.

Format: contains a header row and then row per part. The header row defines a fixed set of initial fields, then columns that correspond with the tests in the “Tests” section.Example:

PartId,PartText,XLoc,YLoc,HardBin,SoftBin,TestTime,TestSite,PF,"101 Cont_In"

1,My first part,-1,-4,1,1,233,1,P,-0.80025625

Keys:

PartId: numeric or text part identifier

Part Text: optional description of the part

XLoc: optional x location on the wafer

YLoc: optional y location on the wafer

HardBin: bin number corresponding with a number in the “Bins” table with BinType HARD

SoftBin: bin number corresponding with a number in the “Bins” table with BinType SOFT

TestTime: time to test this part in milliseconds

TestSite: test site value, corresponding to site value in “Bins” table

PF: P for passed part, F for failed part, empty for unknown