# IEC/TR 62258-8

Edition 1.0    2008-05

# TECHNICAL
# REPORT

Semiconductor die products –
Part 8: EXPRESS model schema for data exchange

IEC/TR 62258-8:2008(E)

## About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

## About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

- Catalogue of IEC publications: www.iec.ch/searchpub

The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub

Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.

- Electropedia: www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv

If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00

![IEC logo]

# IEC/TR 62258-8

Edition 1.0  2008-05

# TECHNICAL
# REPORT

**Semiconductor die products –**
**Part 8: EXPRESS model schema for data exchange**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

PRICE CODE    U

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

————————

## SEMICONDUCTOR DIE PRODUCTS –

## Part 8: EXPRESS model schema for data exchange

## FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.

2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.

3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.

4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.

5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.

6) All users should ensure that they have the latest edition of this publication.

7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.

8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.

9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

The main task of IEC technical committees is to prepare International Standards. However, a technical committee may propose the publication of a technical report when it has collected data of a different kind from that which is normally published as an International Standard, for example "state of the art".

IEC 62258-8, which is a technical report, has been prepared by subcommittee 47: Semiconductor devices.

The text of this technical report is based on the following documents:

| Enquiry draft | Report on voting |
|---|---|
| 47/1927/DTR | 47/1952/RVC |

Full information on the voting for the approval of this technical report can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts in the IEC 62258 series, under the general title *Semiconductor die products,* can be found on the IEC website. Further parts may be added as required.

The committee has decided that the contents of this publication will remain unchanged until the maintenance result date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

A bilingual version of this publication may be issued at a later date.

# INTRODUCTION

This technical report is based on the work carried out in the ESPRIT 4$^{th}$ Framework project GOODDIE which resulted in the publication of the ES 59008 series of European specifications. Organisations that helped prepare this document included the ESPRIT ENCAST project, the Die Products Consortium, JEITA, JEDEC and ZVEI.

**SEMICONDUCTOR DIE PRODUCTS –**

**Part 8: EXPRESS model schema for data exchange**

## 1  Scope

This part of IEC 62258, which is a technical report, has been developed to facilitate the production, supply and use of semiconductor die products, including

- wafers,

- singulated bare die,

- die and wafers with attached connection structures,

- minimally or partially encapsulated die and wafers.

This Technical Report contains an EXPRESS model schema that describes the elements needed for data exchange and that will allow the implementation of the requirements of the IEC 62258-1, IEC 62258-5 and IEC 62258-6 standards, as well as providing an exchange structure that is complementary to those defined in IEC 62258-2. It is also complementary to and compatible with the questionnaire in IEC 62258-4.

## 2  Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 60050 (all parts),   *International Electrotechnical Vocabulary*

IEC 62258-1,   *Semiconductor die products – Part 1: Requirements for procurement and use*

IEC 62258-2,   *Semiconductor die products – Part 2: Exchange data formats*

IEC/TR 62258-4,   *Semiconductor die products – Part 4: Questionnaire for die users and suppliers*

IEC 62258-5,   *Semiconductor die products – Part 5: Requirements for information concerning electrical simulation*

IEC 62258-6,   *Semiconductor die products – Part 6: Requirements for information concerning thermal simulation*

ISO 10303-11:2004, *Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual*

ISO 10303-21:2002, *Industrial automation systems and integration – Product data representation and exchange – Part 21: Implementation methods: Clear text encoding of the exchange structure*

## 3   Terms and definitions

For the purposes of this document, relevant terms, which are defined in IEC 60050 together with additional terms and acronyms as given in IEC 62258-1 apply.

## 4   General

The EXPRESS model schema given in this Technical Report complies with ISO 10303-11 and allows for exchange of data on die devices using the STEP Physical File format (SPF) as defined in ISO 10303-21.

To comply with IEC 62258-1, that standard requires that suppliers of die devices shall furnish information that is necessary and sufficient for users of die devices at all stages of design, procurement, manufacture and test of products containing them. The EXPRESS model schema at Annex A defines an exchange mechanism for structuring such information using a representation that complies with the schema and as such is intended as an aid to compliance with the standard.

Whilst it is expected that much of the information supplied will be in the public domain and available from such sources as manufacturers' data sheets, neither the standard nor the schema places an obligation on a supplier to make information public. Any information that a supplier considers to be proprietary or commercially sensitive may be supplied under the terms of a non-disclosure agreement.

## 5   Data exchange

The EXPRESS model schema at Annex A implements all the entities as defined by IEC 62258-2 for the DDX format. In addition, it includes additional entities extending the range of that data as follows:

- Expansion of data on organisations (manufacturer, supplier etc.) to include addresses and contacts
- Sub-division of some entities to make their values clearer.

Annex B contains an example of a STEP Physical File based on the model schema using a fictitious example similar to that employed in IEC 62258-2, extended to cover additional data requirements. It is possible that software may be available for conversion of data produced using the spreadsheet associated with IEC 62258-4 into this format. In any case, a wide range of tools is available commercially for handling and processing STEP files.

The electronic form of the schema contained in this Technical Report may be downloaded from the IEC website. The copyright conditions applying to the use of the electronic file are those that apply to IEC database standards, which permit the use of such information in electronic form for bona-fide e-commerce but do not permit its sale to third parties or other commercial use.

# Annex A
(normative)

# EXPRESS model schema

## A.1 General

This Annex contains the full EXPRESS listing of the schema, annotated with comments and explanatory text. The order of text in this clause is determined primarily by the order imposed by the EXPRESS language, secondarily by importance.

```
*)
SCHEMA ddx_schema_version_2_0;
(*
```

## A.2 Type definitions

This clause contains definitions for the types used within this EXPRESS model.

```
*)
TYPE date_type = STRING(10) FIXED;
END_TYPE;

TYPE text_type = STRING(255);
END_TYPE;

TYPE geometric_unit_type
= ENUMERATION OF (micron, metre, millimetre, inch, mil);
END_TYPE;

TYPE geometric_view_value
= ENUMERATION OF (top,bottom);
END_TYPE;

(*
**    A    distance    in    the    corresponding    "geometric_unit".
*)
TYPE geometric_value = REAL;
END_TYPE;

TYPE integer_value = INTEGER;
WHERE
  non_negative: SELF >= 0;
END_TYPE;
(*
```

Formal propositions:

 **non_negative:** The integer is non-negative..

```
*)
TYPE angle_value
= INTEGER;
WHERE
  valid_value: {0 <= SELF <= 359};
END_TYPE;
(*
```

Formal propositions:

 **valid_value:** The angle can take values from 0 to 359 degrees..

```
*)
TYPE celsius_value
= REAL;
END_TYPE;

TYPE time_value
= REAL;
END_TYPE;

TYPE watt_value
= REAL;
END_TYPE;

TYPE device_name_type
= TEXT_TYPE;
END_TYPE;

TYPE standards_compliance_type
= text_type;
END_TYPE;

TYPE additional_screening_type
= text_type;
END_TYPE;

TYPE reliability_calculation_type
= text_type;
END_TYPE;

TYPE product_status_type
= text_type;
END_TYPE;

TYPE testability_features_type
= text_type;
END_TYPE;

TYPE additional_test_type
= text_type;
END_TYPE;

TYPE form_of_supply_type
= text_type;
END_TYPE;

TYPE packing_code_type
= text_type;
END_TYPE;

TYPE wafer_die_step_size_type
= size_value;
END_TYPE;

TYPE wafer_gross_die_count_type
= integer_value;
END_TYPE;

TYPE wafer_index_type
= ENUMERATION OF (flat, notch);
END_TYPE;

TYPE wafer_index_orientation_type
= angle_value;
END_TYPE;

TYPE wafer_reticule_step_size_type
= size_value;
```

```
END_TYPE;

TYPE wafer_reticule_gross_die_count_type
= integer_value;
END_TYPE;

TYPE ic_technology_type
= text_type;
END_TYPE;

TYPE data_source_type
= text_type;
END_TYPE;

TYPE data_version_type
= text_type;
END_TYPE;

TYPE block_version_type
= text_type;
END_TYPE;

TYPE function_type
= text_type;
END_TYPE;

TYPE manufacturer_type
= text_type;
END_TYPE;

TYPE pad_metallisation_type
= text_type;
END_TYPE;

TYPE die_passivation_material_type
= text_type;
END_TYPE;

TYPE terminal_material_type
= text_type;
END_TYPE;

TYPE terminal_structure_type
= text_type;
END_TYPE;

TYPE type_number_type
= text_type;
END_TYPE;

TYPE die_name_type
= text_type;
END_TYPE;

TYPE die_semiconductor_material_type
= text_type;
END_TYPE;

TYPE die_back_detail_type
= text_type;
END_TYPE;

TYPE die_substrate_material_type
= text_type;
END_TYPE;

TYPE die_mask_revision_type
```

```
= text_type;
END_TYPE;

TYPE bump_material_type
= text_type;
END_TYPE;

TYPE simulator_compliance_type
= text_type;
END_TYPE;

TYPE simulator_version_type
= text_type;
END_TYPE;

TYPE simulator_name_type
= text_type;
END_TYPE;

TYPE simulator_model_file_type
= text_type;
END_TYPE;

TYPE die_substrate_connection_type
= ENUMERATION OF (CONN, ISOL, OPT, NA, NK);
END_TYPE;

TYPE ellipse_type
= ENUMERATION OF (E);
END_TYPE;

TYPE io_type
    = ENUMERATION OF (I, O, B, G, V, A, N, U, T, X, H, L);
END_TYPE;

TYPE moisture_sensitivity_level_type
    = ENUMERATION OF (1, 2, 2a, 3, 4, 5, 5a, 6);
END_TYPE;
(*
```

## A.3   File structure

This subclause contains the two structures defined for the DDX language, DEVICE and DEVICE_BLOCK.

EXPRESS specification:

```
*)

ENTITY ddx_file;
  devices: SET [1:?] OF device;
  device_blocks: SET [1:?] OF device_block;
END_ENTITY;
(*
```

Attribute definitions:

**devices**: a set of device names.

**device_blocks**: a set of device blocks containing the die data.

## A.4   Device names

This subclause contains the name of a device for inclusion in a set of devices.

EXPRESS specification:

```
*)

ENTITY device;
  device_name: device_name_type;
INVERSE
  containing_file: ddx_file FOR devices;
END_ENTITY;
(*
```

Attribute definitions:

**device_name**: a name by which a device can be identified.

## A.5   Device block

This subclause contains the definition of the contents of a device block.

EXPRESS specification:

```
*)

ENTITY device_block
  ABSTRACT SUPERTYPE OF (ONEOF (bare_die, bumped_die,
    lead_frame_die, minimally_packaged_device));
  described_device: device; -- reference
  block_creation_date: date_type;
  block_version: block_version_type;
  type_number: type_number_type;
  die_packaged_part_name: OPTIONAL type_number_type;
  manufacturer: manufacturer_type;
  function: function_type;
  data_source: data_source_type;
  data_version: data_version_type;
  die_name: die_name_type;
  die_mask_revision: OPTIONAL die_mask_revision_type;
  die_geometry_data: die_geometry;
  material_data: materials;
  terminal_types: SET [1: ?] OF terminal_type;
  connection_count: integer_value;
  terminals: SET [1: ?] OF terminal;
  groups: OPTIONAL SET [2: ?] OF group;
  permutations: OPTIONAL SET [2: ?] of permutable;
  rating_data: ratings;
  wafer_data: wafer;
  quality_and_reliability_data: quality_and_reliability;
  simulation_data: SET [0: ?] OF simulator_data;
  fiducial_types: SET [0: ?] OF fiducial_type;
  fiducials: SET [0: ?] OF fiducial;
  handling_packing_storage_data: OPTIONAL handling_packing_storage;
  die_picture: SET [0: ?] OF document_or_file;
  die_data_file: SET [0: ?] OF document_or_file;
DERIVE
  terminal_type_count: INTEGER:= SIZEOF (terminal_types);
  terminal_count: INTEGER:= SIZEOF (terminals);
INVERSE
  containing_file: ddx_file FOR device_blocks;
```

```
WHERE
  valid_device: described_device IN containing_file.devices;
  terminal_numbers: terminal_type_count <= terminal_count;
END_ENTITY;
(*
```

## A.6    Die geometry

This subclause contains those properties for the geometry of a die.

EXPRESS specification:

```
*)

ENTITY die_geometry;
  geometric_units: geometric_unit_type;
  geometric_view: geometric_view_value;
  size: die_size_and_shape;
  thickness: geometric_value;
  geometric_origin: position;
  x_tolerance: OPTIONAL geometric_tolerance;
  y_tolerance: OPTIONAL geometric_tolerance;
  thickness_tolerance: OPTIONAL geometric_tolerance;
  terminal_x_position_tolerance: OPTIONAL geometric_tolerance;
  terminal_y_position_tolerance: OPTIONAL geometric_tolerance;
END_ENTITY;
(*
```

## A.7    Die size

This subclause contains those properties for the size of a rectangular or elliptical die.

EXPRESS specification:

```
*)

ENTITY die_size_and_shape;
  die_size: size_value;
  shape: OPTIONAL ellipse_type;
END_ENTITY;
(*
```

## A.8    Bare die type

This subclause contains those properties which are specific to non-bumped bare die.

EXPRESS specification:

```
*)

ENTITY bare_die
  SUBTYPE OF (device_block);
UNIQUE
  u1: described_device;
END_ENTITY;
(*
```

## A.9   Bumped bare die type

This subclause contains those properties which are specific to bumped bare die.

EXPRESS specification:

```
*)

ENTITY bumped_die
  SUBTYPE OF (device_block);
  bump_material: text_type;
  bump_shape: text_type;
  bump_size: size_value;
  bump_height: geometric_value;
  bump_height_tolerance: OPTIONAL geometric_tolerance;
  bump_drawing: SET [0: ?] OF document_or_file;
  bump_attachment_method: text_type;
UNIQUE
  u1: described_device;
END_ENTITY;
(*
```

## A.10   Lead-frame die type

This subclause contains those properties which are specific to bare die with attached lead frames.

EXPRESS specification:

```
*)

ENTITY lead_frame_die
  SUBTYPE OF (device_block);
UNIQUE
  u1: described_device;
END_ENTITY;
(*
```

## A.11   Minimally-packaged device

This subclause contains those properties which are specific to minimally-packaged die devices..

EXPRESS specification:

```
*)

ENTITY minimally_packaged_device
  SUBTYPE OF (device_block);
  package_material: OPTIONAL text_type;
  package_style: OPTIONAL text_type;
  connection_type: OPTIONAL text_type;
  seated_height: OPTIONAL geometric_value;
  moisture_sensitivity_level: OPTIONAL
    moisture_sensitivity_level_type;
  package_drawing: SET [0: ?] OF document_or_file;
UNIQUE
  u1: described_device;
END_ENTITY;
```

```
(*
```

## A.12   Die delivery forms

This subclause contains the property defining the form in which die are delivered.

<u>EXPRESS specification:</u>

```
*)

ENTITY die_delivery_form;
  delivery_form: text_type;
END_ENTITY;
(*
```

## A.13   Terminal types

This subclause contains the type and name of a connecting terminal on a die.

<u>EXPRESS specification:</u>

```
*)

ENTITY terminal_type
  ABSTRACT SUPERTYPE OF (ONEOF (rectangle_terminal_type,
    circle_terminal_type,                          ellipse_terminal_type,
polygon_terminal_type));
  terminal_type_name: text_type;
INVERSE
  containing_device_block: device_block FOR terminal_types;
UNIQUE
  name_in_block: terminal_type_name, containing_device_block;
END_ENTITY;
(*
```

## A.14   Rectangular terminal

This subclause contains those properties for a rectangular terminal.

<u>EXPRESS specification:</u>

```
*)

ENTITY rectangle_terminal_type
  SUBTYPE OF (terminal_type);
  rectangle_size: size_value;
END_ENTITY;
(*
```

## A.15   Circular terminal

This subclause contains those properties for a circular terminal.

<u>EXPRESS specification:</u>

```
*)
```

```
ENTITY circle_terminal_type
  SUBTYPE OF (terminal_type);
  diameter: geometric_value;
END_ENTITY;
(*
```

## A.16  Elliptic terminal

This subclause contains those properties for an elliptic terminal.

EXPRESS specification:

```
*)

ENTITY ellipse_terminal_type
  SUBTYPE OF (terminal_type);
  axes: size_value;
END_ENTITY;
(*
```

## A.17  Polygonal terminal

This subclause contains those properties for a polygonal terminal.

EXPRESS specification:

```
*)

ENTITY polygon_terminal_type
  SUBTYPE OF (terminal_type);
  coordinates: LIST [3:?] OF position;
WHERE
  implicitly_closed:
    coordinates[1] <> coordinates [SIZEOF (coordinates)];
END_ENTITY;
(*
```

Attribute definitions:

**coordinates:** the list of coordinate pairs which define the vertices of the polygon. There shall be at least three pairs.

Formal propositions:

**implicitly_closed:** The polygon is closed with a vector from the last vertex to the first vertex which are distinct points.

## A.18  Terminals

This subclause contains those properties which define a terminal.

EXPRESS specification:

```
*)

ENTITY terminal;
  terminal_identifier: text_type;
  corresponding_connection: OPTIONAL text_type;
```

```
  corresponding_terminal_type: terminal_type; -- reference
  location: position;
  orient: orientation;
  terminal_name: OPTIONAL text_type;
  pin_function: OPTIONAL io_type;
INVERSE
  containing_device_block: device_block FOR terminals;
UNIQUE
  number_in_block: terminal_identifier, containing_device_block;
WHERE
  valid_connection:
    corresponding_connection IN containing_device_block.connections;
  valid_terminal_type:
    corresponding_terminal_type IN
      containing_device_block.terminal_types;
END_ENTITY;
(*
```

## A.19  Groups

This subclause contains those terminals or groups which define a group of terminals or of other groups.

<u>EXPRESS specification</u>:

```
*)

ENTITY group;
  group_identifier: text_type;
  terminal_set: OPTIONAL LIST [1:?] of UNIQUE terminal;
  group_set: OPTIONAL LIST [1:?] of UNIQUE group;
INVERSE
  containing_device_block: device_block FOR groups;
UNIQUE
  u1: group_identifier, containing_device_block;
WHERE
  at_least_one: EXISTS (terminal_set) OR EXISTS (group_set);
END_ENTITY;
  (*
```

<u>Attribute definitions</u>:

**group_identifier:** the unique name or code used to identify the group.

**terminal_set:** the set of terminals in the group.

**group_set:** the set of other groups in the group.

<u>Formal propositions</u>:

**at_least_one:** There must be at least one set of terminals or one set of groups.

## A.20  Permutability

This subclause defines those terminals or groups which are permutable.

<u>EXPRESS specification</u>:

```
*)

ENTITY permutable;
  permutation_terminal_set: OPTIONAL LIST [2:?] of UNIQUE terminal;
  permutation_group_set: OPTIONAL LIST [2:?] of UNIQUE group;
WHERE
  at_least_one: EXISTS (terminal_set) OR EXISTS (group_set);
  but_not_both: NOT (EXISTS (terminal_set) AND EXISTS (group_set));
END_ENTITY;
  (*
```

Attribute definitions:

**permutation_terminal_set:** the set of terminals in the permutation.

**permutation_group_set:** the set of groups in the permutation.

Formal propositions:

**at_least_one:** There must be at least one set of terminals or one set of groups.

**but_not_both:** There may not be both a set of terminals and a set of groups.

## A.21  Ratings

This subclause contains those properties that define the limits for operating power and temperature of the die device.

EXPRESS specification:

```
*)

ENTITY ratings;

  maximum_temperature: OPTIONAL celsius_value;
  max_temperature_time: OPTIONAL time_value;
  power_range: OPTIONAL watt_value;
  minimum_operation_temperature: OPTIONAL celsius_value;
  maximum_operation_temperature: OPTIONAL celsius_value;

END_ENTITY;

  (*
```

## A.22  Quality and reliability

This subclause contains those properties for quality and reliability of the die device.

EXPRESS specification:

```
*)

ENTITY quality_and_reliability;
  outgoing_quality_level: OPTIONAL outgoing_quality;
  standards_compliance: OPTIONAL standards_compliance_type;
  additional_screening: OPTIONAL additional_screening_type;
  reliability_data: OPTIONAL reliability;
  product_status: OPTIONAL product_status_type;
```

```
   testability_features: OPTIONAL testability_features_type;
   additional_test_requirements: OPTIONAL additional_test_type;
END_ENTITY;
   (*
```

## A.23  Quality level

This subclause contains those properties defining the outgoing quality of a device.

EXPRESS specification:

```
*)

ENTITY outgoing_quality;
   value: REAL;
   units: text_type;
   description: text_type;
   electrical_parameter_conditions: text_type;
END_ENTITY;
(*
```

## A.24  Reliability

This subclause contains those properties defining the reliability figure for a device, together with the assocxiated units and the conditions under which it was obtained.

EXPRESS specification:

```
*)

ENTITY reliability;
   value: REAL;
   units: text_type;
   conditions: text_type;
   reliability_calculation: OPTIONAL document_or_file;
END_ENTITY;
(*
```

## A.25  Handling, packing, storage and assembly

This subclause contains those properties concerning handling, packing and storage of the die device.

EXPRESS specification:

```
*)

ENTITY handling_packing_storage;
   die_delivery_forms: SET[0: ?] OF die_delivery_form;
   packing_code: OPTIONAL packing_code_type;
   assembly_information: OPTIONAL text_type;
   special_requirements: OPTIONAL text_type;
   attach_methods_and_materials: OPTIONAL text_type;
   bonding_method_and_materials: OPTIONAL text_type;
   attachment_limitations: OPTIONAL text_type;
   process_limitations: OPTIONAL text_type;
   storage_limitations: OPTIONAL text_type;
END_ENTITY;
   (*
```

## A.26  Material data

This subclause contains those properties for materials with which the die device is fabricated.

EXPRESS specification:

```
*)

ENTITY materials;
  ic_technology: OPTIONAL ic_technology_type;
  die_semiconductor_material: OPTIONAL
    die_semiconductor_material_type;
  die_substrate_material: OPTIONAL die_substrate_material_type;
  substrate_connection: die_substrate_connection;
  pad_metallisation: OPTIONAL pad_metallisation_type;
  die_passivation_material: OPTIONAL die_passivation_material_type;
  die_back_detail: OPTIONAL die_back_detail_type;
  terminal_material: OPTIONAL terminal_material_type;
  terminal_structure: OPTIONAL terminal_structure_type;
END_ENTITY;
  (*
```

## A.27  Wafer data

This subclause contains those properties for the wafer on which a die device is fabricated.

EXPRESS specification:

```
*)

ENTITY wafer;

  wafer: OPTIONAL wafer_size;
  wafer_thickness: OPTIONAL geometric_value;
  wafer_thickness_tolerance: OPTIONAL geometric_tolerance;
  wafer_die_step_size: OPTIONAL wafer_die_step_size_type;
  wafer_gross_die_count: OPTIONAL wafer_gross_die_count_type;
  index: OPTIONAL wafer_index;
  wafer_reticule_step_size: OPTIONAL wafer_reticule_step_size_type;
  wafer_reticule_gross_die_count: OPTIONAL
    wafer_reticule_gross_die_count_type;

END_ENTITY;

  (*
```

## A.28  Wafer size

This subclause contains those properties defining the size of a wafer

EXPRESS specification:

```
*)

ENTITY wafer_size;
  diameter: geometric_value;
  units: geometric_unit_type;
END_ENTITY;
(*
```

## A.29  Wafer index

This subclause contains those properties defining the type and position of an index on a wafer

EXPRESS specification:

```
*)

ENTITY wafer_index;
  index_type: wafer_index_type;
  index_orientation: wafer_index_orientation_type;
END_ENTITY;
(*
```

## A.30  Simulation data

This subclause contains those properties for simulation data and the associated simulator.

EXPRESS specification:

```
*)

ENTITY simulator_data;
  simulator_type: text_type;
  model_file: simulator_model_file_type;
  file_date: OPTIONAL date_type;
  simulator_name: simulator_name_type;
  simulator_version: simulator_version_type;
  minimum_compliance_level: OPTIONAL simulator_compliance_type;
  terminal_set: OPTIONAL SET [1:?] of terminal;
  group_set: OPTIONAL SET [1:?] of group;
INVERSE
  containing_device_block: device_block FOR simulation_data;
UNIQUE
  data_for_simulator: simulator_type, containing_device_block;
END_ENTITY;

  (*
```

## A.31  Fiducial type

This subclause contains those properties defining the appearance of a fiducial on a die.

EXPRESS specification:

```
*)

ENTITY fiducial_type;
  fiducial_type_name: text_type;
  fiducial_file_name: text_type;
  fiducial_size: size_value;
INVERSE
  containing_device_block: device_block FOR fiducial_types;
UNIQUE
  name_in_block: fiducial_type_name, containing_device_block;
END_ENTITY;
(*
```

## A.32  Fiducial

This subclause contains those properties defining the position and orientation of a fiducial on a die.

EXPRESS specification:

```
*)

ENTITY fiducial;
  fiducial_number: integer_value;
  corresponding_fiducial_type: fiducial_type; -- reference
  location: position;
  orient: orientation;
INVERSE
  containing_device_block: device_block FOR fiducials;
UNIQUE
  number_in_block: fiducial_number, containing_device_block;
WHERE
  valid_fiducial_type:
    corresponding_fiducial_type IN
      containing_device_block.fiducial_types;
END_ENTITY;

(*
```

## A.33  Die and feature size

This sub-clause contains those properties defining length and width of the die or of a feature on the die.

EXPRESS specification:

```
*)

ENTITY size_value;
  x_size: geometric_value;
  y_size: geometric_value;
INVERSE
  containing_device_block: device_block FOR size;
END_ENTITY;

(*
```

## A.34  Position

This subclause contains those properties defining the position coordinates of a terminal or a fiducial on a die.

EXPRESS specification:

```
*)

ENTITY position;
  x: geometric_value;
  y: geometric_value;
INVERSE
  containing_device_block: device_block FOR geometric_origin;
END_ENTITY;

(*
```

## A.35 Orientation

This subclause contains those properties defining the orientation of a terminal or a fiducial on a die.

EXPRESS specification:

```
*)

ENTITY orientation;
  rotation: angle_value;
  mirror_x: BOOLEAN;
  mirror_y: BOOLEAN;
END_ENTITY;
(*
```

## A.36 Date

This subclause contains the definition of dates.

EXPRESS specification:

```
*)

ENTITY date;
  ISO_date: date_type;
END_ENTITY;
(*
```

## A.37 Tolerances

This subclause contains the definition of geometric tolerances.

EXPRESS specification:

```
*)

ENTITY geometric_tolerance;
  neg_tolerance: geometric_value;
  pos_tolerance: geometric_value;
END_ENTITY;
(*
```

## A.38 Substrate connection

This subclause contains those properties defining the connection requirement for the die substrate.

EXPRESS specification:

```
*)

ENTITY die_substrate_connection;
  connection_reqirement: die_substrate_connection_type;
  connection_point: OPTIONAL text_type;
END_ENTITY;
(*
```

## A.39  Document or file type

This subclause contains those items needed to define a document or file containing information about the device.

<u>EXPRESS specification</u>:

```
*)

ENTITY document_or_file;
  document_name: OPTIONAL text_type;
  file_name: OPTIONAL text_type;
  version: OPTIONAL text_type;
  version_date: OPTIONAL date;
  file_type: OPTIONAL text_type;
  mime_type: OPTIONAL text_type;
END_ENTITY;

END_SCHEMA;

(*
*)
```

# Annex B
## (informative)

# STEP physical file example

This example is based on a fictitious device and illustrates many of the features of the schema.

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('DDX/STEP physical file example'),'1.2.1');
FILE_NAME('74ACT00.spf','2006-02-18T12:32:53',('DDXconvert'),( ), '','','D
E Radley');
FILE_SCHEMA(('ddx_schema_version_2_0'));
ENDSEC;

DATA;

#1=DDX_FILE((#2), (#3));
#2=DEVICE('74ACT00');
#3=BARE_DIE(#2, #4, '1.0', '74ACT00', $, 'Fuzziwuzz Logic Ltd', 'Quad two-
input advanced CMOS NAND gate',
'GOOD-DIE database', '1.0', 'Mask T', $, #11, #12, (#20), 14, (#30, #33,
#36, #39, #42, #45, #48, #51,
#54, #57, #60, #63, #66, #69), (#80, #81, #82, #83, #84, #85, #86, #87),
(#90, #91, #92, #93, #94),
#13, #14, #15, (#100), (#110), (#111), #16, $, $);
#4=DATE('2006-02-18');
#5=DIE_SIZE_AND_SHAPE(#6, $);
#6=SIZE_VALUE(1067, 1143);
#7=POSITION(0,0);
#8=DIE_SUBSTRATE_CONNECTION(.CONN., 'Vcc');
#9=DIE_DELIVERY_FORM('Die');
#10=DIE_DELIVERY_FORM('wafer');
#11=DIE_GEOMETRY(.MICRON.,.,TOP., #5, 356, #7, #22, #23, #24, #25, #26);
#12=MATERIALS('CMOS', 'silicon', $, #8, 'Al', $, $, $, $);
#13=RATINGS(150, $, $, -70, 150);
#14=WAFER(#17, 356, $, $, $, #18, $, $);
#15=QUALITY_AND_RELIABILITY(#19, 'IECQ', 'None', #27, 'No changes planned',
$, $);
#16=HANDLING_PACKING_STORAGE((#9, #10), $, $, $, $, $, 'Maximum 175Cel for
2s', $, 'Six months at 25Cel and 50%RH');
#17=WAFER_SIZE(8,.INCH.);
#18=WAFER_INDEX(.FLAT., 90);
#19=OUTGOING_QUALITY(10, 'ppm', '25Cel', 'Vcc=5V');

#20=RECTANGLE_TERMINAL_TYPE('PADR1', #21);
#21=SIZE_VALUE(97, 97);

#22=GEOMETRIC_TOLERANCE(-20, 20);
#23=GEOMETRIC_TOLERANCE(-20, 20);
#24=GEOMETRIC_TOLERANCE(-10, 10);
#25=GEOMETRIC_TOLERANCE(-5, 5);
#26=GEOMETRIC_TOLERANCE(-5, 5);

#27=RELIABILITY(100, 'FIT', 'Temperature 25Cel and mobile phone
application', #28);
#28=DOCUMENT_OR_FILE('MIL HDBK 217E', $, $, $, $, $);

#30=TERMINAL( 'T_1', 1, #20, #31, #32, 'A1', 'I');
#31=POSITION(-385, 422);
#32=ORIENTATION(0,.F.,.,.F.);
#33=TERMINAL( 'T_2', 2, #20, #34, #35, 'B1', 'I');
#34=POSITION(-385, 176);
#35=ORIENTATION(0,.F.,.,.F.);
#36=TERMINAL( 'T_3', 3, #20, #37, #38, 'Y1', 'O');
```

```
#37=POSITION(-385, 11);
#38=ORIENTATION(0,.F.,.F.);
#39=TERMINAL( 'T_4', 4, #20, #40, #41, 'A2', 'I');
#40=POSITION(-385, -236);
#41=ORIENTATION(0,.F.,.F.);
#42=TERMINAL( 'T_5', 5, #20, #43, #44, 'B2', 'I');
#43=POSITION(-208, -423);
#44=ORIENTATION(0,.F.,.F.);
#45=TERMINAL( 'T_6', 6, #20, #46, #47, 'Y2', 'O');
#46=POSITION(-43, -423);
#47=ORIENTATION(0,.F.,.F.);
#48=TERMINAL( 'T_7', 7, #20, #49, #50, 'GND', 'G');
#49=POSITION(123, -423);
#50=ORIENTATION(0,.F.,.F.);
#51=TERMINAL( 'T_8', 8, #20, #52, #53, 'Y3', 'O');
#52=POSITION(385, -423);
#53=ORIENTATION(0,.F.,.F.);
#54=TERMINAL( 'T_9', 9, #20, #55, #56, 'B3', 'I');
#55=POSITION(385, -166);
#56=ORIENTATION(0,.F.,.F.);
#57=TERMINAL( 'T_10', 10, #20, #58, #59, 'A3', 'I');
#58=POSITION(385, -1);
#59=ORIENTATION(0,.F.,.F.);
#60=TERMINAL( 'T_11', 11, #20, #61, #62, 'Y4', 'O');
#61=POSITION(385, 164);
#62=ORIENTATION(0,.F.,.F.);
#63=TERMINAL( 'T_12', 12, #20, #64, #65, 'B4', 'I');
#64=POSITION(385, 423);
#65=ORIENTATION(0,.F.,.F.);
#66=TERMINAL( 'T_13', 13, #20, #67, #68, 'A4', 'I');
#67=POSITION(38, 423);
#68=ORIENTATION(0,.F.,.F.);
#69=TERMINAL( 'T_14', 14, #20, #70, #71, 'VCC', 'V');
#70=POSITION(-129, 423);
#71=ORIENTATION(0,.F.,.F.);

#80=GROUP ( 'G_1', (#30, #33), () );
#81=GROUP ( 'G_2', (#39, #42), () );
#82=GROUP ( 'G_3', (#54, #57), () );
#83=GROUP ( 'G_4', (#63, #66), () );
#84=GROUP ( 'G_5', (#36), (#80) );
#85=GROUP ( 'G_6', (#45), (#81) );
#86=GROUP ( 'G_7', (#51), (#82) );
#87=GROUP ( 'G_8', (#60), (#83) );

#90=PERMUTABLE ( (#30, #33), () );
#91=PERMUTABLE ( (#39, #42), () );
#92=PERMUTABLE ( (#54, #57), () );
#93=PERMUTABLE ( (#63, #66), () );
#94=PERMUTABLE ( (), (#84, #85, #86, #87) );

#100=SIMULATOR_DATA('SPICE', 'SP74ACT00.MOD', '1997-09-17', 'pSPICE',
  '4.0.1', '2G6', (#30, #33, #39, #42, #54, #57, #63, #66), $);
#101=SIMULATOR_DATA('SPECTRE', 'SP74ACT00.S', '1998-11-05', 'Spectre',
  '4.2.1 1992', '2G6, Level-3', $, (#84, #85, #86, #87));

#110=FIDUCIAL_TYPE('fiduc1', '74ACT00FD1.jpg', #112);
#111=FIDUCIAL(1, #110, #113, #114);
#112=SIZE_VALUE(72, 55);
#113=POSITION(-470, 352);
#114=ORIENTATION(0,.F.,.F.);

ENDSEC;
END-ISO-10303-21;
```