HYDAC INTERNATIONAL

MATCH
Project Definition Tool

Functional Safety
SIL 2
PL d
AgPL d (SRL 2)

Special features

- **SIL 2 / PL d / AgPL d certified software**
- Software project definition at vehicle and machine level
- Project support for multiple controllers
- Automatic code generation for controllers and displays
- Safety-related development cycle in acc. with V-model supported
- Capturing of requirements, as well as use case and test case management
- Generation of project and development documentation using your own templates
- Comprehensive definition of:
  - CAN communication
  - Error messages and responses
  - Parameter and option lists in NvMem and flash memory
  - I/O pins in compliance with safety requirements
- Use of certified driver modules to control connected actuators and sensors
- Generation of embedded code for supported controller hardware, as well as code for simulating the application on the PC (software in the loop)
- Support of module and integration tests
- Toolbox interface for flexibly incorporating certified library modules

Description

The **P**roject **D**efinition **T**ool (PDT) is a piece of PC software used to develop applications (embedded software) for complex vehicle and machine controls comprising one or more controllers and / or displays.

The PDT is essentially made up of:

- A graphic interface for data entry and maintenance,
- Auto Code Builder for controller and display software, and
- HYDAC MATCH Core software and a standard toolbox.

Three basic versions of the PDT are available:

- **Programmer C:** PDT package for programming individual controllers in “C”, for developers working alone
- ***Programmer CODESYS*:** PDT package for programming individual controllers and displays in CODESYS for developers working alone
- **Standard Developer:** PDT package for system-software development of entire machines performed by developer teams, system specialists and requirement managers

You use the PDT interface in a safety-related development cycle at vehicle level for the following work steps:

- Definition of requirements and software specifications
- Creation of the system design with controllers and displays
- Configuration of the PINs and CAN messages, as well as a cross-vehicle error management
- Configuration of library modules (blocks) for controlling the connected sensors and actuators
- Set-up of the database for parameters, teach values and options

The use of tested and certified toolboxes simplifies and significantly speeds up application development. Once the system has been defined, the PDT generates the MATCH software framework from the project – depending on the programming language – with the certified Auto Code Builder (according to SIL2, PL d and AgPL d). This software framework forms the basis of application development, where the developer can focus on nonstandard functions of the machine.

The PDT also generates complete project documentation from the inputs made and provides the interfaces to the MST (**M**achine **S**ervice **T**ool) maintenance tool and to the TSE (**T**est and **S**imulation **E**nvironment) test tool.

* In preparation

EN 18.532.0/03.18

Technical data

Software system requirements		
Supported operating systems	Windows® 7, 8 or 10 (32-/64-bit)	
Other software	.NET 4.6 Framework, C-Compiler for corresponding controllers PDF display program such as Adobe Acrobat® Reader®	
Hardware system requirements		
Processor	Minimum dual-core processor with 1.6 GHz	
RAM memory requirements	Minimum 2 GB (4 GB or more recommended)	
Hard drive memory requirements	Minimum 1 GB available memory	
Screen resolution	Minimum 1,024 x 768	
Complies with the following standards		
Functional safety	IEC 61508 – SIL 2, EN ISO 13849 – PL d, ISO 25119 & EN 16590 – AgPL d (SRL 2)	
Auto code generation		
Controllers	Programming language C	
Display	Programming language CODESYS 3.5	
Controllers	Non-safety	Safety
	HY-TTC 30-H, HY-TTC 30-I, HY-TTC 32-H, HY-TTC 50, HY-TTC 60	HY-TTC 30S-H, HY-TTC 32S-H, HY-TTC 90, HY-TTC 94, HY-TTC 71, HY-TTC 77, HY-TTC 510, HY-TTC 540, HY-TTC 580
Displays	Touchscreen	Without touchscreen
	HY-eVision² 7.0 and HY-eVision² 10.4	HY-eVision² 7.0

Basic version model code

MATCH PDT – XX – G10 – 000

Program variant

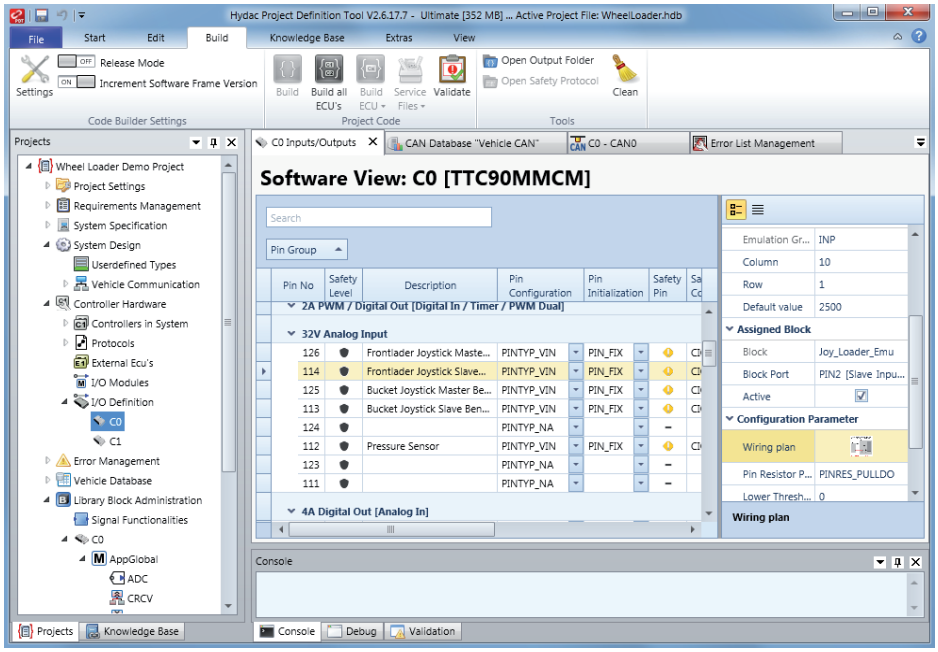
00 = Standard Developer
CP = Programmer C
CD = Programmer CODESYS

Software version

G10 = Current version

Modification number

000 = Standard



Screenshot of PDT pin configuration

Scope of delivery*

Installation program for:

- PDT desktop application,
- MATCH Core for the controllers and displays, as well as PC simulation,
- Standard libraries,
- Standard toolbox, and
- Integrated development environment (Eclipse)

Accessories

(Not included in the scope of delivery; please order separately)

- Cable harnesses for controllers
- Programming cable (CAN) part number: 6149786
  for HY-TTC 50/60/90/94 or
- Programming cable (CAN) part number: 61499787
  for HY-TTC 77
- PCAN dongle ZBS PCAN USB connector
  Part number: 6163719
- Maintenance and test tools (MST, TSE)

* Depending on option and output format ordered

PDT add-ons

An add-on is an extension of the PDT's range of functions.

PDT add-on model code

MATCH PDT AddOn – XX – G10 – YYYY – 000

Program variant

00 = Standard Developer
CP = Programmer C
CD = Programmer CODESYS

Software version

G10 = current version

Extension variant

CEN = Automatic Code Generation of embedded C-Code
CCG = C Code Generation
S2Pd = Safety Certified Code Generation (SIL 2, PL d, AgPL d)
DCD3 = V²-MATCH Library – CODESYS 3.x Code Generation
CANO = CANopen Stack
SPCK = Specification Packet
RCAP = Requirements Capturing
ISOB = ISOBUS Plug-in
UDSB = UDS Basic – Basic UDS Communication Interface
ISYM = SYM-File Import / Export Interface
RQIF = Requirements ReqIF interface
DBCE = DBC export
DMSY = Document Management System
HMG4 = HMG 4000 Integration

Modification number

000 = standard

The table below shows the add-ons available for the different program versions. The corresponding part numbers have been entered for available options. Unavailable options are marked with "–" and add-ons already included are marked with "✓".

Performance characteristics	Standard Developer	Programmer C	Programmer C
Multiple controllers	✓	9576 (MECU)	9645 (MECU)
Safety-certified code (SIL 2, PL d, AgPL d)	✓	9509 (CEN)	9641 (S2Pd)
C Code Generation	9588 (CCG)	✓	–
Display library V²-MATCH – CODESYS 3.5 Display Code	9501 (DCD3)	9577 (DCD3)	✓
General project documentation as PDF	✓	✓	✓
Document management	✓	9579 (DMSY)	9646 (DMSY)
Extended project specifications in customised design	9503 (SPCK)	–	–
Requirements Management	9505 (RQIF)	–	–
Requirements ReqIF interface	9507 (RCAP)	–	–
"UDS Basic" diagnostic interface	✓	9511 (UDSB)	9642 (UDSB)
CANopen Stack	9502 (CANO)	9510 (CANO)	✓
Icon file import/export interface	✓	9512 (ISYM)	9647 (ISYM)
ISOBUS plug-in	9508 (ISOB)	–	–
Start pages editor	✓	–	–
Multilingual information output	✓	–	–
DBC export	9569 (DBCE)	9580 (DBCE)	9648 (DBCE)
HMG 4000 Integration	✓	9513 (HMG4)	9643 (HMG4)

Toolboxes

A toolbox is a compilation of library modules. It is made up of blocks and signal elements. All modules are certified according to the afore-mentioned functional safety standards.

Signal elements

Signal elements are software modules that generally provide basic functions for your application. They are completely encapsulated and may contain state variables. You can use signal elements in “C” code much like complex variables. You initialise the signal elements with access functions and can thus adjust or use them in the application.

Blocks

Blocks are added in the PDT and generated using the Auto Code Builder with the MATCH software framework. Blocks can be connected to controller input / output pins within the PDT.

Some blocks read one or more input pins (drivers for sensors, switches, etc.), some control one or more output pins (drivers for specific actuators) and some are used for functions such as closed-loop control without hardware access.

Input blocks enable the frequency, current or voltage to be measured, for example, with appropriate error detection and diagnostic options.

Output blocks control the likes of different valve types, lamps and other electric actuators, whereby the relevant diagnostic functionalities are integrated here too.

Every block can:

- Include error detection and thus be connected to the error management system
- Be connected to one or more input / output pins or no pin
- Use parameters connected to database lists
- Build on signal elements.

MATCH toolbox model code

MATCH ToolBox – XXX – G10 – YYY – 000

Controller range

ALL = All supported controller platforms
030 = HY-TTC 30-H, HY-TTC 30S-H
032 = HY-TTC 32-H, HY-TTC 32S-H
050 = HY-TTC 50, HY-TTC 60
071 = HY-TTC 71
077 = HY-TTC 77
090 = HY-TTC 90, HY-TTC 94
500 = HY-TTC 510, HY-TTC 540, HY-TTC 580

Software version

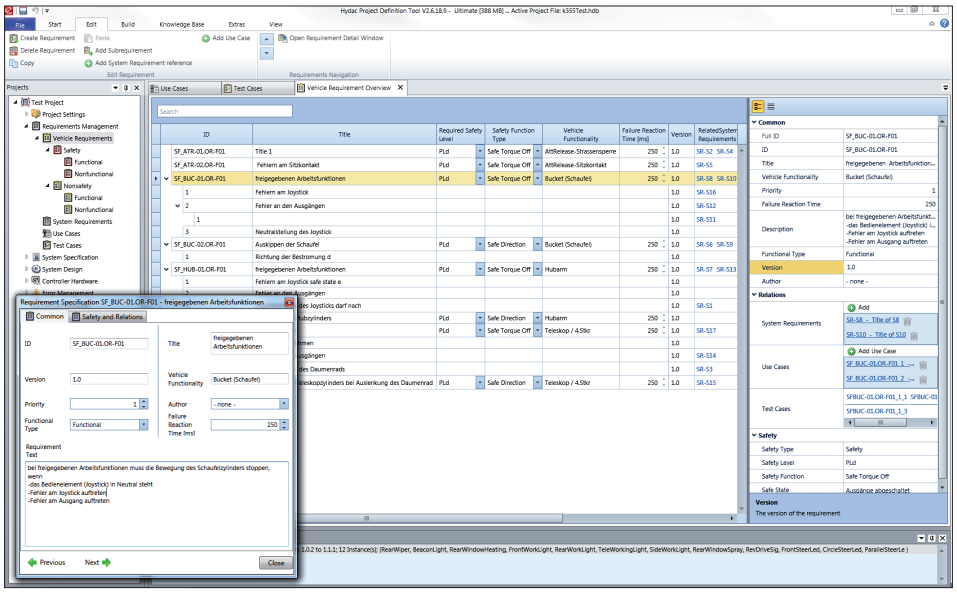
G10 = Current version

Toolbox variant

SCT = Signal Control Toolbox,
further toolboxes on request

Modification number

000 = Standard



Definition of machine requirements with the PDT

Note

The information in this brochure relates to the operating conditions and applications described.

For applications or operating conditions not described, please contact the relevant technical department.

Subject to technical modifications and corrections.

HYDAC ELECTRONIC GmbH
Hauptstr. 27
66128 Saarbrücken, Germany
Tel. +49 6897 509-01
Fax +49 6897 509-1726
E-mail: electronic@hydac.com
Internet: www.hydac.com