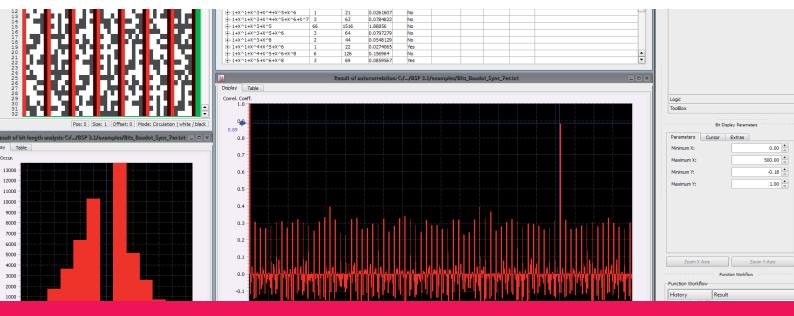


Bit Stream Analysis and Processing



go2ANALYSE is a user friendly and powerful tool for the analysis of unknown signals on bit streams level.

Key facts

- Powerful offline bit stream analysis tool
- Check unknown bit streams for known/existing decoders
- Identify unknown coding details or parameters
- Analyse existing decoders
- Process generic bit streams

- Search for periodical and non-periodical patterns
- Demultiplexing and deinterleaving
- Use of DDL decoders
- Several alphabets and user-definable code tables
- Recording, saving and replay of analysis steps



WORKFLOW

	DEMODULATION	VISUALIZATION		UNKNOWN SIGNALS	EDIT
	DECODING	VOICE LISTENING	KNOWN SIGNALS	CODE	COMPILATION
	RECORDING	TEXT	PARAMETER	BIT STREAM	DEBUGGING
	SIGNAL PROCESSING	RESULTS	MANUAL ANALYSIS	INDEPTH CODE ANALYSIS	DECODER DEVELOPMENT
				974	
LIVE STRE					

go2ANALYSE

Offline analysis, manipulation of bit streams to determine the code characteristics.

- Wide range of logical, statistical, demultiplexing, deinterleaving, LFSR, search and binary modulations functions
- Adapt or modify functions by applying a scripting language
- Use of DDL decoders
- Record, save and replay analysis steps
- Write specific test programms in order to identify unknown codings (e.g. CRC-polynoms)
- Program parts used for the code analysis can be used in the resulting decoders
- Easy implementation of libraries and use of external programms
- Processing of already conditioned bit streams

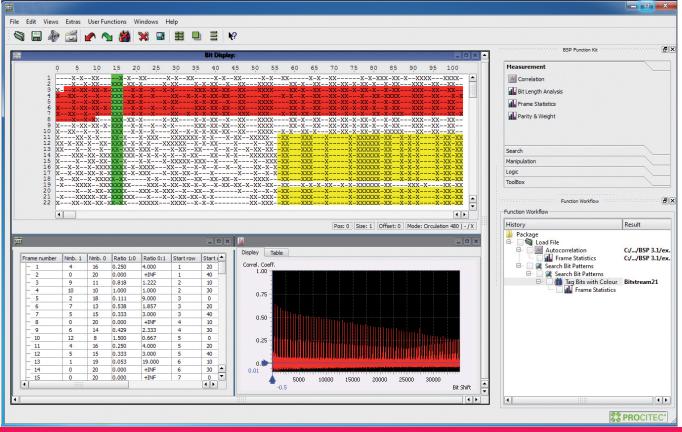


Analysis tool for specialists

Where the work of manual signal analysis ends, go2ANALYSE enables coding specialists to gather the basis information for the modification of existing or the writing of new decoders.

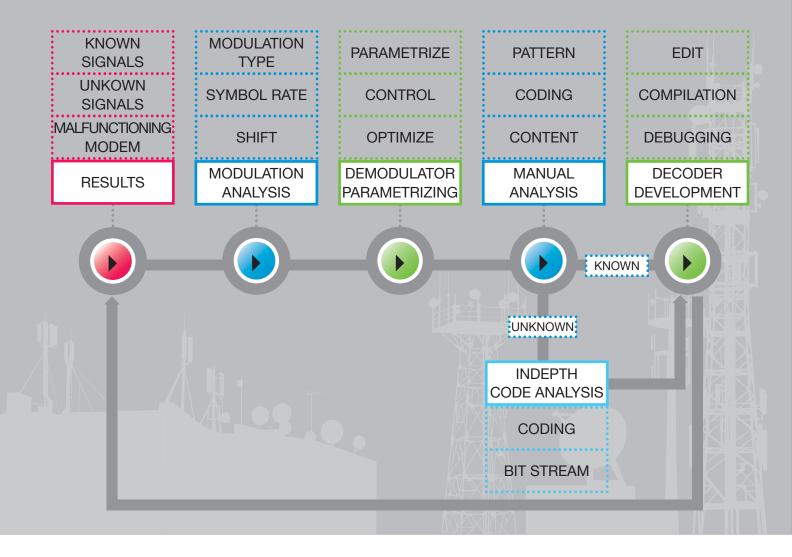
go2ANALYSE offers a wide range of statistical, mathematical and manipulative functions to determine the characteristics of the applied coding, combined with vital features such as bit stream visualizations in various formats, logic operations and editing functions. For manipulating standardised binary raw data are required. go2ANALYSE facilitates the analysis, providing functions to record, save and replay the analysis steps. Further, existing DDL decoders can be applied to the bit stream currently processed, and the code tables and alphabets in use are accessible for modifications. This way the analysts' knowledge and experience gets build in for future automatic processing.

go2ANALYSE is intended for users familar with the theory of coding, demodulation and error correction as well as the respective know-how in mathematics and algorithms.



Bit stream analysis with a modern and user-definable GUI

USE CASES:



Use Case

Indepth analysis of unknown protocols

Modern radio monitoring systems support the operators in many ways, but when it comes to new unknown modem types, the operators reach their limits.

Often, specialists and technical experts have to bring up their entire know-how and spend hours of exhausting manual analysing in order to convert their work in a performing decoder.

The result of the bit stream analysis is a userdefined chronology of functions/commands – ideally resulting in a complete and precise decoding. Using so called "Analysis Decoders", bit streams can gradually be analysed, visualised and in the last step decoded. Without additional go2SIGNALS tools like go2DECODE or go2MONITOR these "Analysis Decoders" can be used directly.

By using programming languages (e.g. DDL, C, C++) the required know-how regarding commands and functions can be implemented in decoders and used in manual or automated radio monitoring systems.

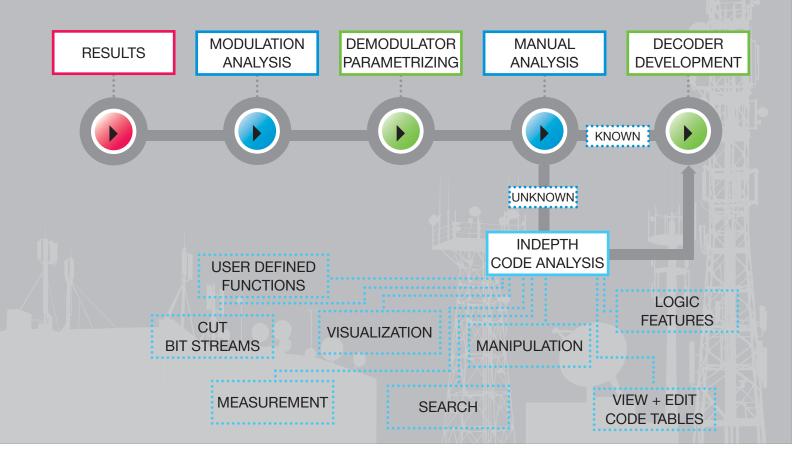


0	5	10	15	20	25	30	٠		30	35	40	45	50	55 🖉
2 XXX 4 X-2 4 X-2 5 6 XX2 5 6 XX2 7 B D 1 X 2 XX2 - XX2 4 - XX2 4 - XX2 4 - XX2 4 - XX2 4 - XX2 4 - XX2 2 XX2 2 XX2 4 - XX2 2 - XX2 2 - XX2 4 - XX2 2		<pre></pre>	- XX - X - X - X - X X - X X - X X - X X - X X - X - X - X - X - X - X - X - X - X - X - X - X	××	-XXXX XXX -XXX XXX -XXXX -XXXX -XXXX -XXX -XXX -XXX -XXX -XXX -XXX -XX- -XXXX	X X X X X X X X X		1 2 3 4 5 6 7 8 9 10 11 12	X X X X X	-XX X-X X-X X-X X-X X-X X-X X-X X-X (XXX-X (X-X	-XXX -XX-XX -XXX -XX-XX -XX-XX -XX-XX	<pre></pre> <pre><pre><pre><pre><pre><pre><pre><</pre></pre></pre></pre></pre></pre></pre>	XX XX XX XXX XX XX XX XX XX XX	XXX XXX XXX XXX XXX XXX XXX XXX XXX XXX
1 - 2 - 3 - 4 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - 5	- X X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X - X X X X X X X X X X X X X	-XX-X XXX -XXX XXX XX XX- XX- XX- XX- XX- XX- XX- XX- XX-X- X-X-X-	XX- X-XX- XXXX- XXXX- X-XX- X-XX- X-XX- X-XX- X-XX- X-X-X -X-XX- -X-XX- -X-XX- -X	XX XX XX 	XXX- XX-XX- -XXX XX- XX-X X- XX-X X- XXX-X- X XXX-X- X XXX-X- X XXX-X- X XXX XX XXX		X-X -XXX X-X -X-XX- -XX-X -XX-X -XX-X -XX-X -XX-X -XX-X -X-X XXX-X -X-X XXX-X -X-X -X-X XXX-X -X-X	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	SUEDW 5, SI SUEDW WEST SCHNE FISCE WEST DOGGE WEST	BIS NOR ERE SCHN ER: UM 7, R CHWEISE	HT: WEST 7, CHNEEBOE E NORDSE , NORDDR ANFANGS DWEST 7 EEBOEEN, ASCH NOR	EN, ANE E: EHEND, DIESIG BIS 8, SEE 3 DWESTDF	NORDWES FANGS DI ETWAS A S, SEE 2 LANGSAM BIS 4 M REHEND,	ESIG, S BNEHMEN BIS 4 1 ABNEHM METER. SPAETER
	11111	23456789		X - X - X - X X - X	1,0 x x x - x - x - x -	× - × × - × × - × × - × × - × × - × × - ×				2 3 4 5 6 7 8 9 0 1 1 2 3 1112 3	X - - X - X - X XX X - X X - X - X X -			

- ① Bit visualization with symbol quality allows to focus on areas of low bit error rate.
- ② Highlighting of differences
- ③ Using an alphabet a decoded bit stream is outputted as readable text.
- ④ Bitwise Exclusive OR (XOR) operation of two bit stream files. The different bits are marked.

.

FUNCTIONS



Functions:

Bit stream visualization and navigation

Our software go2ANALYSE provides the analyst with all necessary functionality to visualise, evaluate and process the bit stream. A large number of measurement, search, manipulation and logic features simplifies the analysis process.

User defined functions

Furthermore, even more complex problems in the bit stream analysis can be solved as go2ANALYSE is an open tool. It gives you the possibility by using the scripting language DDL (Decoder Description Language) to enhance or to modify existing functions.

Some of the go2ANALYSE functions have been realised that way. The source code is part of the shipment and provides the basis for specific modifications.

Required parameter input masks are being established via XML-data in a simple syntax. "Analysis Decoders" which are generated this way offer not only text output but also modified bit streams and simple graphic pictures.

Standard programming interface

An integrated programming interface (C++, etc.) offers additional expandability. This way self-developed algorithms and decoders can be embedded; logfiles and even speech outputs are possible. External libraries and programs can be integrated easy and simple.

Command and analysis history

The entire workflow is documented in single steps. It can be reproduced step-by-step at any time. Interim results can be shown at any step.



Specifications overview Data acquisition Tool-based bit arow packed bits arow packed bits arow packed bits are packed bits arow packed bits a							
Bracked brany Bracked brany Localization Explait: Others on request Documentation Explait: Others on request Decommended Moi. Irel 15 2 Core, 2 GHz, min. 4 CB FAML 16 GB mecommerded PC hardware Moi. Irel 16 2 Core, 2 GHz, min. 4 CB FAML 16 GB mecommerded PC hardware Second mechanism in the second method is the second m	Specifications overview						
Lacalization English: Others on request Opcomonitation PDF Volme Hold PC Folline Hold PC Foline Hold PC Folline Hold PC Foline Hold PC Folline Ho	Data acquisition	Packed binary					
Documentation PDF User manual / PDF Online-Holp Recommended Min. Hall FS 2.Core, 2 Polt, zmi. 4.08 RAM, 16 GB recommended PC hardware HDD min. 50 GB recommended (stepands on binary file input) So Windows 7 / 10 64 ht Features Encommended Software Feature Remarks Bil Stream Visualization x ² , L/H, 71 instaud of 1/0 Alignment: Burst/Circulation length Sy 2 Dial Stream Visualization x ² , L/H, 71 instaud of 1/0 Alignment: Burst/Circulation length Bil Stream Visualization x ² , L/H, 71 instaud of 1/0 Alignment: Burst/Circulation length Bil of isol Symool Stream Str	Localization						
Paceware Min. Inst is 2 core. 2 cHz. min. 4 dB PAM, 16 GB recommended sources of the source of t							
PC hardware HDD: min. 30 GB recommended (sippands on biamy file input) OS Windows 7 / 10 G4 bit Features Remarks Software Feature Remarks Bit Stream Visualization Angoment: Burst/Cinculation length Circulation length Algoment: Burst/Cinculation length Circulation length Bit Stream Visualization Software Features Automatic search for non-periodic sequences Bit Stream Visualization Automatic search for non-periodic sequences Repeated patterns Analysis Autocorrelation Remarks Resource of two bitstreams Analysis Autocorrelation Repeated patterns Reference of two bitstreams Analysis Autocorrelation Repeated patterns Resource of two bitstreams Manipulation / Transformation Bit ength = matysis Mark start, stop and parity bits Mark start, stop and parity bits Toto for LFSR Analysis and handling of linear loedback shift neglisters Bits Hits quality attomatics Bits Hits quality attomatics Binary Modulation Mark Start stop and parity bits Mark start stop and parity bits Mark start stop and parity bits Binary Modulation Mark start stop and parity bits Mark start stop and parity bits Mark start stop and		•					
Peakures Peakures Software Feature Remarks Alignment: Burst/Circulation length Crut size changeable Bit Stream Visualization Fort size changeable Cut/Cony/Paste Undo/Redo Graution length Bit offsol Tigs bits with different colors Show difference of two bitsreams Analysis Automatic search for non-periodic sequences Crosscorelation Automatic search for non-periodic sequences Analysis Automatic search for periodic sequences Repeated patients Manipulation / Transformation Decimation Remarks Of ratio Decimation Cuting United Streams Mark Stat, stop and parity bits Decimation Decimation Cuting Of ratio Decimation Cuting Decimation Decimation Cuting United Streams Decimation Cuting Nore Kob Streams Decimation Cuting United Streams Decimation Cuting Deriver Streams Decimation Cuting Nore Kob Streams Decimation Cuting Deriver Streams Decimation <td></td> <td>HDD: min. 50 GB recommended (depends on binary file inp</td> <td></td>		HDD: min. 50 GB recommended (depends on binary file inp					
Software Feature Remarks Algenetic Bit Stream Visualization N., L.H., J. instead of 1/0 Graphical bit display Algenetic But SufCirculation length Curl/Orapy/Paste Und/ORado Bit Stream Visualization For size changeable Graphical bit display Automatic search for non-periodic sequences Reported patterns Analysis Corescorrelation Bit length analysis Automatic search for non-periodic sequences Reported patterns Manipulation / Transformation Decinetation Bit length analysis Reported patterns Manipulation / Transformation Decinetation Decinetation Logic: AND, OR, NOT, XOR selected bits, Logic: AND, OR, NOT, XOR selected bits, Logic: CAND, OR, NOT, XOR selected bits, Logic: CAND, OR, NOT, XOR selected bits, Binary Modulation BIPH-M NRZ-M NRV bitsmeams Map Bits to Text Magnetic and the restance Assesse Binary Modulation NRZ-M NRZ-M NRV Bitsmeams Map Bits to Text Destinetation BiH-M Manchester Predefined code tables: e.g. ASCIB, Baudot, Budot-SSIII-CVR, LEX, Morse, ITA2P User defined code tables: Binary Modulation Map Bits to Text Destinetation BiH-M BiH-K-S BiH-M Manchester Predefined code tables: e.g. ASCIB, Baudot, Budot-SSIII-CVR, LEX, Morse, ITA2P User defined code tables: Binary Modulation Map Bits to Text Destinetation BiH-M BiH-K-S BiH-M BiH-K-S Predefined code tables: Budot-SSIII-CVR, LEX, Morse, ITA2P User defined code t	OS	Windows 7 / 10 64 bit					
Software Feature Remarks Algenetic Bit Stream Visualization N., L.H., J. instead of 1/0 Graphical bit display Algenetic But SufCirculation length Curl/Orapy/Paste Und/ORado Bit Stream Visualization For size changeable Graphical bit display Automatic search for non-periodic sequences Reported patterns Analysis Corescorrelation Bit length analysis Automatic search for non-periodic sequences Reported patterns Manipulation / Transformation Decinetation Bit length analysis Reported patterns Manipulation / Transformation Decinetation Decinetation Logic: AND, OR, NOT, XOR selected bits, Logic: AND, OR, NOT, XOR selected bits, Logic: CAND, OR, NOT, XOR selected bits, Logic: CAND, OR, NOT, XOR selected bits, Binary Modulation BIPH-M NRZ-M NRV bitsmeams Map Bits to Text Magnetic and the restance Assesse Binary Modulation NRZ-M NRZ-M NRV Bitsmeams Map Bits to Text Destinetation BiH-M Manchester Predefined code tables: e.g. ASCIB, Baudot, Budot-SSIII-CVR, LEX, Morse, ITA2P User defined code tables: Binary Modulation Map Bits to Text Destinetation BiH-M BiH-K-S BiH-M Manchester Predefined code tables: e.g. ASCIB, Baudot, Budot-SSIII-CVR, LEX, Morse, ITA2P User defined code tables: Binary Modulation Map Bits to Text Destinetation BiH-M BiH-K-S BiH-M BiH-K-S Predefined code tables: Budot-SSIII-CVR, LEX, Morse, ITA2P User defined code t	Features						
Fort size changeable CurVCopy/Paste Circulation length Bits with quality Symbolis of bits Bits with quality Symbolis of bits Symbolis of bits Analysis Autocorrelation Autocorrelation Repeated patterns Bit length analysis Marks status Manipulation / Transformatio Beinterleaving Inversion: Mirror / NOT Decimation Curtical Curtical Decimation Curtical Curtical Automatic search for periodic sequences Bertherns Bit difference of two bits Manipulation / Transformatio Decimation Curtical Decimation Curtical Presented patterns Statutonal Constraints Decimation Curtical Decimation Curtical Presented patterns Statutonal Constraints Decimation Curtical Decimation Curtical Presented patterns Statutonal Constraints Decimation Curtical Decimation Curtical Presented patterns Bits oftest Statutonal Berleaving Vortedow NR2-M Bits ofte		Remarks					
ConstructionRepeated patternsBit length analysisMark start, stop and parity bitsOf ratioBit length analysisOf natioDeinterfieseManipulation / TransformatioDeinterfieseManipulation / TransformatioDeinterfieseDemultiplexingUnversion: Mirror / NOTDecimationCuttingDemultiplexingUnversion: Mirror / NOTDecimationCuttingTools for LFSRAnalysis and handling of linear feedback shift registersBinary ModulationNR2-MNR2-SBiPH-I. ManchesterBinary ModulationNR2-MNR2-SBiPH-I. ManchesterManagementComplete workflow recordedManagementDesplay and parity bitsManagementDisplayed as tree of commands and results Undo/Redo (several steps)Save/Load workflowOpen selected bits in external tool (configurable)VerkflowMemManagementDecoder (becompleted)Decoder DevelopmentImeBasic functionsApply completed software decoders to a loaded bitstream Use of DL decoders (the Decoder Description Language is a programming language for the implementati- on of software decoders)Function libraryPreprocessing Symbol conversions Decoder can supply different output types such as bitstream output, graphic output, marker output, pro- gress bar and text outputFunction libraryPreprocessing Symbol conversions Decoder con supply different output types such as dub-routines (special functions on request)Percein libraryReprocessing Symbol co	Bit Stream Visualization	Font size changeable Graphical bit display Circulation length Bit offset Tag bits with different colors	Cut/Copy/Paste Undo/Redo Bits with quality				
LargerDecimationCutting Viterbi correction Descrambling DestuffingTools for LFSRAnalysis and handling of linear feedback shift registers Bertekamp-Massey Linear complexitiesBIPH-M Baudot-3Shift-CVR, HEX, Morse, ITA2P User defined code tables: e.g. ASCIB, Baudot, Baudot-3Shift-CVR, HEX, Morse, ITA2P User defined code tables Save/Load workflow recorded Displayed as tree of commands and results Undor/Redo (several steps) Save/Load workflow Save/Load workflow Uhad/Redo (several steps) Save/Load workflow Uhad/Redo (several steps) Save/Load workflow Use of DDL decoders (the Decoder Description Language is a programming language for the implementati- on of software decoders to a loaded bitstream output, graphic output, mor- gress bar and text output Pre-procedures CHC, Harming, Viterbi, BCH, Reed-Solomon Elementary arithmetic and bit manipulations Table handling Pre-procedures Channel selection Pre-procedures Channel selection Pre-procedures Channel selection Pre-procedures Channel selection Pre-procedures Channel selection Pre-procedures Channel selec	Analysis	Crosscorrelation Bit length analysis 0/1 ratio	Repeated patterns Mark start, stop and parity bits Testing against codes: Hamming, Reed-Solomon,				
Beriekamp-Massey Linear complexitiesBirth-M BIPH-M BIPH-L BIPH-L MRZ-S BIPH-L ManchesterBIPH-M BIPH-SMap Bits to TextNRZ-S BIPH-L ManchesterBIPH-M BIPH-SMap Bits to TextNSB/LSB Normal/Inversepredefined code tables: e.g. ASCIB, Baudot, 	Manipulation / Transformation	Decimation Demultiplexing Logic: AND, OR, NOT, XOR selected bits,	Cutting Viterbi correction Descrambling				
NRZ-S BIPH-L ManchesterBIPH-SMap Bits to TextMSB/LSB Normal/Inversepredefined code tables: e.g. ASCII8, Baudot, Baudot-3Shift-CYR, HEX, Morse, ITA2P User defined code tablesWorkflowComplete workflow recorded Displayed as tree of commands and results Undo/Redo (several steps) Save/Load workflowReplay saved workflow with different bitstreams Change command parameters in workflow delete individual commandsIntegrate External ToolsOpen selected bits in external tool (configurable)TemDecoder DevelopmentItemSave/Load workflowBasic functionsApply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language is a programming language for the implementati- on of software decoders) Decoder can supply different output types such as bitstream output, graphic output, marker output, pro- 	Tools for LFSR	Berlekamp-Massey					
Normal/InverseBaudot-3Shift-CYR, HEX, Morse, ITA2P User defined code tablesWorkflowComplete workflow recorded Displayed as tree of commands and results Undo/Redo (several steps) Save/Load workflowReplay saved workflow with different bitstreams Change command parameters in workflow delete individual commandsIntegrate External ToolsOpen selected bits in external tool (configurable)Complete workflow recorded User FunctionsUser FunctionsItemItemBasic functionsApply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language is a programming language for the implementati- on of software decoders) Decoder can supply different output types such as bitstream orgress bar and text outputCheck and correction procedures: CRC, Hamming, Viterbi, BCH, Reed-Solomon Elementary intentie and bit manipulations Table handlingFunction libraryPre-processing Symbol conversions Descrambling procedures CHannel selections Pattern search Burst detection Pattern search Burst detection Content related help Syntax highlightingDecoder EditorAutomatic command completion CompletCompilerGeneration of binary decoder files	Binary Modulation	NRZ-S					
ManagementDisplayed as tree of commands and results Undo/Redo (several steps) Save/Load workflowChange command parameters in workflow delete individual commandsIntegrate External ToolsOpen selected bits in external tool (configurable)Total commandsUser FunctionsDecoder DevelopmentItemBasic functionsApply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language for the implementati- on of software decoders) Decoder can supply different output types such as bitstream output, graphic output, marker output, pro- gress bar and text outputFunction libraryPre-processing Symbol conversions Descrambling procedures Channel selection Pattern search DeinterleavingCheck and correction procedures: CRC, Hamming, Viterbi, BCH, Reed-Solomon Elementary arithmetic and bit manipulations Table handling Branches and sub-routines (special functions on request)Decoder EditorAutomatic command completion Content related help Syntax highlightingSprach Branches and sub-routines (special functions on request)CompilerGeneration of binary decoder filesSprach Branches and sub-routines (special functions on request)	Map Bits to Text		Baudot-3Shift-CYR, HEX, Morse, ITA2P				
User Functions Decoder Development Item Basic functions Apply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language is a programming language for the implementati- on of software decoders) Decoder can supply different output types such as bitstream output, graphic output, marker output, pro- gress bar and text output Function library Pre-processing Symbol conversions Descrambling procedures Channel selections Channel selections Descrambling procedures Deinterleaving Check and correction procedures: CRC, Harming, Viterbi, BCH, Reed-Solomon Elementary arithmetic and bit manipulations Table handling Pattern search Burst detection Forward/backward time jumps Deinterleaving Decoder Editor Automatic command completion Content related help Syntax highlighting Compiler Generation of binary decoder files		Displayed as tree of commands and results Undo/Redo (several steps)	Change command parameters in workflow delete				
Decoder DevelopmentItemBasic functionsApply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language is a programming language for the implementati- on of software decoders) Decoder can supply different output types such as bitstream output, graphic output, marker output, pro- gress bar and text outputFunction libraryPre-processing Symbol conversions Descrambling procedures Channel selections Pattern search Burst detection Forward/backward time jumps DeinterleavingCheck and correction procedures: CRC, Hamming, Viterbi, BCH, Reed-Solomon Branches and sub-routines (special functions on request)Decoder EditorAutomatic command completion Content related help Syntax highlightingAutomatic coder files	Integrate External Tools	Open selected bits in external tool (configurable)					
Decoder DevelopmentItemBasic functionsApply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language is a programming language for the implementati- on of software decoders) Decoder can supply different output types such as bitstream output, graphic output, marker output, pro- gress bar and text outputFunction libraryPre-processing Symbol conversions Descrambling procedures Channel selections Pattern search Burst detection 	User Functions						
Basic functions Apply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language is a programming language for the implementation of software decoders) Decoder can supply different output types such as bitstream output, graphic output, marker output, progress bar and text output Function library Pre-processing Check and correction procedures: Symbol conversions CRC, Hamming, Viterbi, BCH, Reed-Solomon Descrambling procedures Elementary arithmetic and bit manipulations Channel selections Table handling Pattern search Branches and sub-routines (special functions on request) Forward/backward time jumps Deinterleaving Decoder Editor Automatic command completion Compiler Generation of binary decoder files		Item					
Function libraryPre-processingCheck and correction procedures: CRC, Hamming, Viterbi, BCH, Reed-Solomon Elementary arithmetic and bit manipulations Table handling Pattern search Burst detection Forward/backward time jumps DeinterleavingCheck and correction procedures: CRC, Hamming, Viterbi, BCH, Reed-Solomon Elementary arithmetic and bit manipulations Table handlingDecoder EditorAutomatic command completion Content related help Syntax highlightingCompilerCompilerGeneration of binary decoder filesElementary arithmetic and bit manipulations Table handling Branches and sub-routines (special functions on request)	•	Apply compiled software decoders to a loaded bitstream Use of DDL decoders (the Decoder Description Language on of software decoders) Decoder can supply different output types such as bitstre	e is a programming language for the implementati-				
Content related help Syntax highlighting Compiler Generation of binary decoder files	Function library	Pre-processing Symbol conversions Descrambling procedures Channel selections Pattern search Burst detection Forward/backward time jumps	CRC, Hamming, Viterbi, BCH, Reed-Solomon Elementary arithmetic and bit manipulations Table handling Branches and sub-routines (special functions on				
		Content related help Syntax highlighting					
	Compiler						



... monitoring a connected world

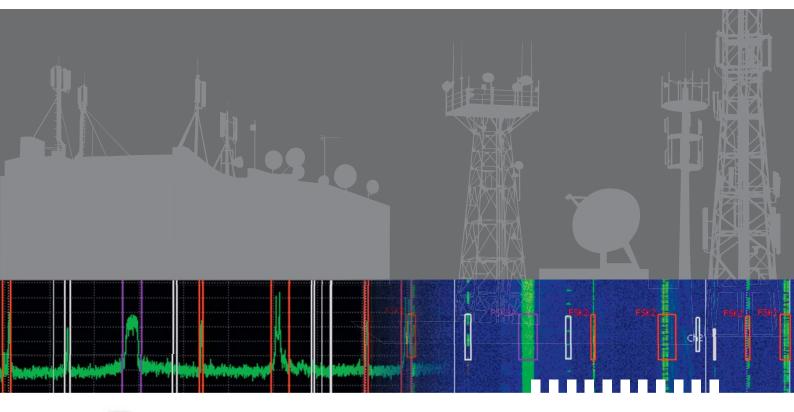
PROCITEC GmbH

Rastatter Strasse 41 75179 Pforzheim Germany

Phone: +49 7231 155 61-0 Fax: +49 7231 155 61-11

Email: sales@procitec.de Further information on www.go2signals.de www.procitec.de







17.2 01/2018 (Subject to modification)