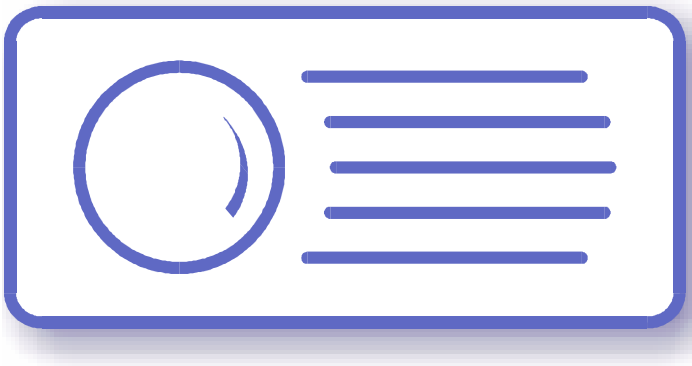


Tecnoroll BMB srl

Via Olona 9, 21040 Vedano Olona VA

www.tecnoroll.it

FLEXMod-711

module

Connectors, pin description and programming protocol

Rev A, August, 30th 2012

Hardware revision 2.00

Firmware version 0.06

1. The use of this product in conformity with the DVB Standard is prohibited in any case, unless the license of patents applicable to the DVB patent group is granted.
2. It's the purchaser's duty to check which patents with potentially payable author's rights are used in his/her own final project and act accordingly with the bodies in charge.
3. The contents of this manual are subject to change without notice.
4. Of the specifications described in this manual, the functions and performance the supply of which is assured with this product are only those that can be confirmed with the separate Basic Firmware attached.
5. When the product or information described in this manual is used, our company does not guarantee or grant any license of the intellectual property rights and other rights of our company or of third parties. If any problems concerning the ownership rights of third parties attributable to the above use occur, our company will not be responsible for them.
6. Since the product described in this manual was not developed specially for equipment related to human life, those who intend to use it for equipment whose failure or malfunction may directly threaten human life or endanger the human body or for special applications requiring high reliability (such as aerospace, nuclear power control, submarine repeaters, travel control, and medical equipment for life support) must consult our sales responsible personnel in advance. Our company will not be responsible for any damages that may occur due to such use without consulting us.
7. Although our company is endeavoring to improve quality and enhance reliability, semiconductor products will fail with a certain probability. Users are requested to pay due attention to safe design techniques such as redundant design, preventive measures against the spreading of fire, over-current, and malfunctions, so that failure of the products described in this manual will not result in accidents leading to injury or death, fire, or social damage.
8. The product described in this manual is not designed to be radiation-resistant.
9. This board is just an 'electronic component' that cannot work in a standalone mode, therefore all the tests and qualifications for electromagnetic compatibility, all the CE certifications and any other regulations concerning the final product implemented by the customer are to be carried out by the customer himself referring to the final product for which this electronic component has been used.

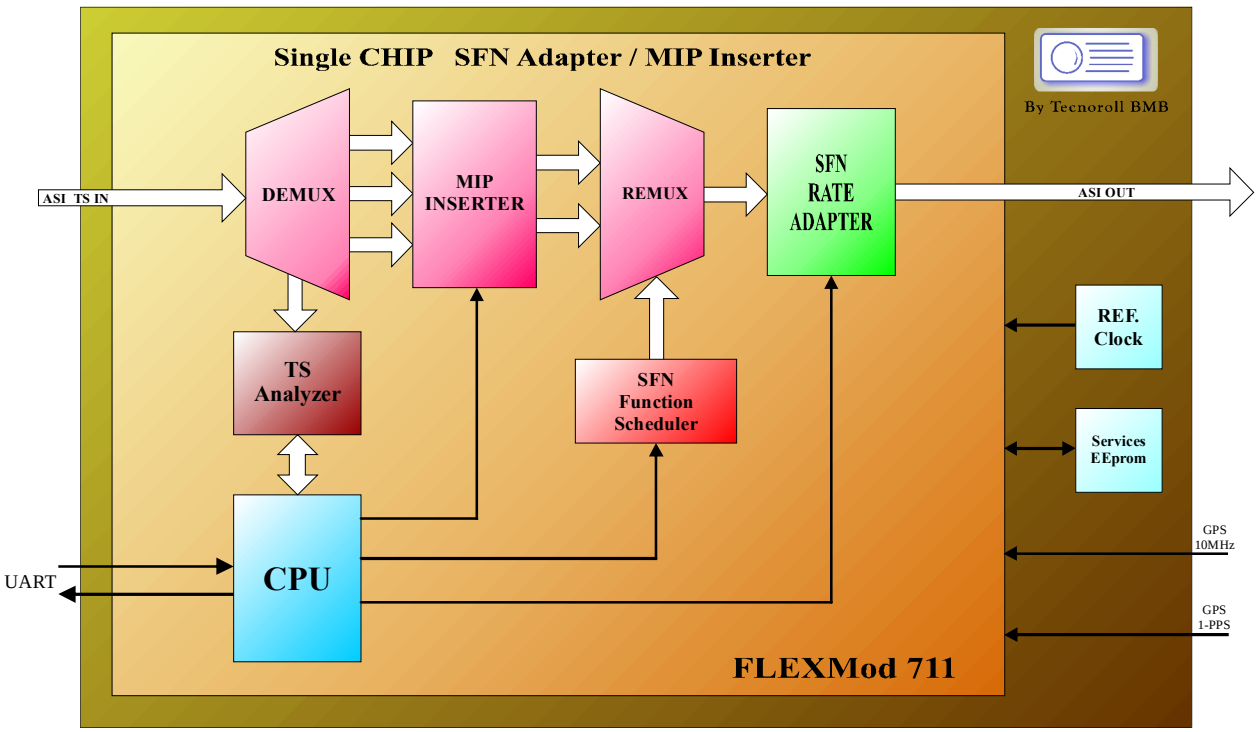
Revision history:

Rev A, August, 30th 2012, abesani@tecnoroll.it
First release

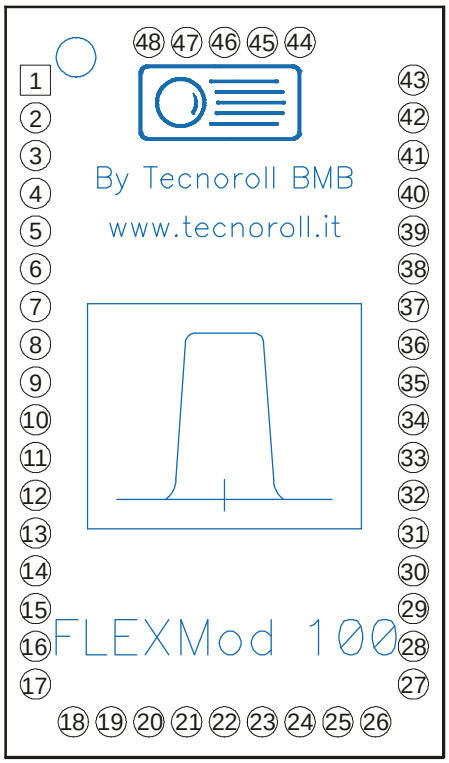
Index

1. FLEXMod Block Diagram

1.1. FLEXMod-711



2. Connectors and pin description



FLEXMod-711, Top view

Pin	Symbol	Type	Description
1	GND	P	Digital Ground
2	1V8	P	+1.8V Digital Power Supply
3	1V8	P	+1.8V Digital Power Supply
4	GND	P	Digital Ground
5	RSVD	-	Reserved
6	ASI_IN_1	I_{LVDS+}	LDVS TS ASI Input #1. Requires a Cable equalizer and an adaptation network. See *ASI Input* on page 25 .
7		I_{LVDS-}	
8	ASI_IN_2	I_{LVDS+}	LDVS TS ASI Input #2. Requires a Cable equalizer and an adaptation network. See *ASI Input* on page 25 .
9		I_{LVDS-}	
10	RSVD	-	Reserved
11	RSVD	-	Reserved
12	RSVD	-	Reserved
13	RSVD	-	Reserved
14	RSVD	-	Reserved
15	10MHz_OUT	O_3	10Mhz reference output
16	PPS_OUTPUT	O_3	1 Pulse-per-second reference output
17	REF_10MHz	I_3	10Mhz reference input

Pin	Symbol	Type	Description
18	GND	P	Digital Ground
19	1V2	P	+1.2V Digital Power Supply
20	3V3	P	+3.3V Digital Power Supply
21	TxD	O_3	RS232 TX Line
22	RxD	I_3	RS232 RX Line
23	PPS_INPUT	I_3	1 Pulse-per-second reference clock input (GPS).
24	3V3	P	+3.3V Digital Power Supply
25	1V2	P	+1.2V Digital Power Supply
26	GND	P	Digital Ground
27	Flag7	O_3	Flags bit 7
28	Flag6	O_3	Flags bit 6
29	Flag5	O_3	Flags bit 5
30	Flag4	O_3	Flags bit 4
31	Flag3	O_3	Flags bit 3
32	Flag2	O_3	Flags bit 2
33	Flag1	O_3	Flags bit 1
34	Flag0	O_3	Flags bit 0
35	FLAGS_CLK	O_3	Flags SPI Clock Signal (1)
36	FLAGS_DATA	O_3	Flags SPI Data Signal (1)
37	ASI_ OUT	O_{LVDS-}	LDVS TS ASI Output. Requires an adaptation network and a Cable Driver. See *ASI Output* on page 26 for more information.
38		O_{LVDS+}	
39	FLAGS_LD	O_3	Flags SPI Load Signal (1)
40	GND	P	Digital Ground
41	3V3	P	+3.3V Digital Power Supply
42	3V3	P	+3.3V Digital Power Supply
43	GND	P	Digital Ground
44	GND	P	Digital Ground
45	RSVD	-	Reserved
46	GND	P	Digital Ground
47	RSVD	-	Reserved
48	GND	P	Digital Ground

See Appendix A on page 31 for *type* description.
(1) See *Flags SPI sample schematic* on page 27

3. Serial port usage

The Rs232 serial port allows the configuration and the operation of the FLEXMod boards.
This serial port normally operates at 115200 bps, 8 data bits, 1 stop bit, no parity but the operating baud rate can be changed using the *Baud* command. Regardless of the setting done with the *Baud* command, the FLEXMod-711 will always boot using a baud rate of 115200 bps.

This serial port normally echoes back to the terminal the characters received. When this is not desirable (because you are using a microcontroller, for example) echoing can be disabled using the *ECHO* command (see *Echo* on page 10).
Most of the command used to set values with one or more parameters, returns the current configured value if issued without any parameter.

A list of available commands can be obtained using the *HELP* command.

3.1. HELP

Used for:	FLEXMod Help.
Parameters:	None
Example:	Help ↵

4. FLEXMod system configuration

The configuration setup of the operating parameters is stored in an onboard EEPROM. Commands change the current operating parameters in ram: to make any configuration change permanent, the *SAVE* command (see below) should be issued.

4.1. Save

Used for:	Saves operating parameters into EEPROM.
Parameters:	None
Example:	Save↵
Notes:	-.

4.2. Clear

Used for:	Clears EEPROM values to factory defaults.
Parameters:	None
Example:	Clear↵
Notes:	The factory defaults will be loaded and the system rebooted.

4.3. Reboot

Used for:	Restart the FLEXMod.	
Parameters:	Option	
Example:	Reboot↵	*(Reboot both FPGA and microprocessor)*
	Reboot 0↵	*(Reboot microprocessor only)*
Notes:	-.	

4.4. Baud

Used for:	Changes the serial port baud rate.
Parameters:	BaudRate
Example:	Baud 9600↵ Baud 115200↵
Notes:	Boot baud rate will always be 115200. Nearly all baud rates can be selected up to 3Mbps.

4.5. Echo

Used for:	Activates/Deactivates character echoing.
Parameters:	0 or 1
Example:	Echo 0↵ Echo 1↵
Notes:	Echoing could be disabled to ease the use of a microcontroller. When operating the FLEXMod using a terminal program, having the FLEXMod echoing the characters back simplifies its use.

4.6. HexMode

Used for:	Activates/Deactivates hexadecimal replies.

Parameters:	0 or 1
Example:	HexMode 0↵ HexMode 1↵
Notes:	Hexadecimal replies could be enabled to ease the processing by a microcontroller.

4.7. GetFWVersion

Used for:	Query FLEXMod product name and firmware version.
Parameters:	None
Example:	GetFWVersion↵
Notes:	Return value is: *FlexMOD 711 - SFN Adapter* *x Version:x.xx*

4.8. GetSN

Used for:	Query FLEXMod serial number.
Parameters:	None
Example:	GetSN↵
Notes:	Return value is: *SN: xx.xx.xx.xx.xx.xx.xx*

4.9. Welcome

Used for:	Activates/Deactivate welcome message.
Parameters:	0 to 1
Example:	Welcome 0↵ *(Welcome message disabled)* Welcome 1↵ *(Standard welcome message)*
Notes:	There is a special welcome message sent by the FLEXMod when receiving the first ENTER (CR, 0x0D) character after a boot. This is helpful when using the FLEXMod connected to a USB serial port since the original power-up message will be lost because the USB is not yet connected. It is advisable to disable this welcome message when operating the FLEXMod using a microcontroller.

4.10. GetTemp

Used for:	Display currently FlexMOD temperature.
Parameters:	None
Example:	GetTemp↵
Notes:	The temperature is shown in Celsius degree.

4.11. TempAlarm

Used for:	Set point for output alarm signal.
Parameters:	30 to 125
Example:	TempAlarm 75↵ *(Set the alarm temperature to 75 °C)*
Notes:	Temperature is set in Celsius degrees. When FlexMOD temperature is above this set value the OverTemp bit on the *Flags* SPI (see on page 27) will be signaled.

5. Single Frequency Network (SFN) setup commands

5.1. TPSParms

Used for: Set or get TPS (modulator) parameters.
Parameters: CONST,FEC,CARR,GUARD,CHBW,PRI,ALPHA,INT,SLI,MPE
Example: TPSParms 2,1,1,3,1,1,0,0,0,0↵ *(64QAM, FEC 2/3, Guard Interval 1/4, 8K, High priority TS, Alpha 0, Native interleaver, Time slicing not used, MPE-FEC not used).*
TPSParms ,2,,,,,,,↵ *(Change FEC to 3/4 only)*
Notes: Parameters are as follow:

CONST	constellation: 0=QPSK, 1=16QAM, 2=64QAM
FEC	code rate: 0=1/2, 1=2/3, 2=3/4, 3=5/6, 4=7/8
CARR	transmission mode: 0=2K, 1=8K, 2=4K
GUARD	guard interval: 0=1/32, 1=1/16, 2=1/8, 3=1/4
CHBW	bandwidth: 0=7Mhz, 1=8Mhz, 2=6MHz, 3=5MHz
PRI	0=Low priority TS, 1=High priority TS/non-hierarchical
ALPHA	0=Non-hierarchical, 1=alpha 1, 2=alpha 2, 3=alpha 4
INT	0=Native, 1=in-depth interleaver
SLI	0=Time slicing not used, 1=Time slicing used
MPE	0=MPE-FEC not used, 1=MPE-FEC used

You can omit any parameter that does not change.

5.2. MIPMaxDelay

Used for: Set or get Network maximum delay.
Parameters: 0-99999999
Example: MIPMaxDelay 5000000↵ *(Set the network maximum delay to 0.5s)*

5.3. MIPPointer

Used for: Set or get MIP pointer configuration.
Parameters: PER,PT
Example: MIPPointer 1,0↵ *(Periodic pointer, set to 0)*
MIPPointer 0↵*(Aperiodic pointer, pointer value -PT- is ignored, if present)*
Notes: Periodic pointer means that the pointer is always set to the programmed value (PT). In case the programmed values is higher than the MegaFrame size, the pointer value is set to 0.
Aperiodic pointer means that the pointer changes every megaframe. The programmed value (PT) is therefore ignored.

5.4. FrequencyError

Used for: Set or get maximum allowable time drift in nSec.
Parameters: VCXO,PPS
Example: FrequencyError 25000,1000↵

Notes: This command limits the allowable time drift for either the internal VCXO oscillator or the PPS reference. In case this limits are exceeded, the module operation is halted until a new stable lock can be achieved.

5.5. PPsWatchDog

Used for: Set or get GPS PPS watchdog control.
Parameters: 0 or 1
Example: PPsWatchDog 0↵ *(Disable PPS Watchdog)*
PPsWatchDog 1↵ *(Enable PPS Watchdog)*
Notes: If the PPS Watchog is enabled, loss of the PPS signal will stop the output.
If it is disabled, the output will continue even if the PPS signal is lost. If the 10MHz reference is adequate, the output will be correct even if the PPS signal is no longer present.

5.6. TestMode

Used for: Enable the operation of the MIP inserter without a 10MHz reference or PPS signal.
Parameters: MODE,PPS,VAL
Example: TestMode 0↵ *(Normal operation mode. Both 10MHz and PPS inputs are required)*
TestMode 1↵ *(Both 10MHz and PPS signals are internally generated)*
TestMode 2↵ *(The 10MHz input is required while the PPS signal is internally generated)*
TestMode 3,,2000↵ *(Both 10MHz and PPS signals are internally generated, but the tune value for the 10MHz is explicitly given)*
Notes: PPS is the duty cycle of the generated PPS, 10 to 90 (10% to 90%). Default is 50 (50%) and if not given the last used value is kept.
VAL is the tuning value for *TestMode 3* operation. Range is 0 to 4095.

6. MIP Functions configuration commands

6.1. MIPFuncEnable

Used for: Add/Modify/Delete an Enable Function List.
Parameters: TXId,Func[,Func,Func]
Example: MIPFuncEnable 77, 0,2,4↵ *(Start transmitting the enable function, to the transmitter with an ID of 77, enabling function. Tags 0, 2 and 4).*
Notes: If the *MIPFuncEnable* command is given with the *TXId* parameter only, the Enable function list for this TXId, if present, is removed and no longer present in the MIP.
For a complete list of the function tags, refer to the *ETSI TS 101 191 Technical Specification*.

6.2. MIPFuncBandwidth

Used for: Add/Modify/Delete a Bandwidth Function.
Parameters: TXId,Bandwidth,WaitForEnable
Example: MIPFuncBandwidth 77,0,0↵ *(Start transmitting a 5MHz (0) bandwidth, function to the transmitter with an ID of 77. Does not set the wait_for_enable_flag).*
Notes: The bandwidth function is used for a modulator bandwidth of 5MHz. 6, 7 and 8MHz are programmed via the TPS parameters directly.
If the *wait_for_enable_flag* is set, the SFN modulator will wait for an *enable* command before applying this function. See the *MIPFuncEnable* command.
If the *MIPFuncBandwidth* command is given with the *TXId* parameter only, the Bandwith Function for this TXId, if present, is removed and no longer present in the MIP.

6.3. MIPFuncCellID

Used for: Add/Modify/Delete a CellID Function.
Parameters: TXId,CellID,WaitForEnable
Example: MIPFuncCellID 77,99,0↵ *(Start transmitting a CellID = 99, function to the transmitter with an ID of 77. Does not set the wait_for_enable_flag).*
Notes: If the *wait_for_enable_flag* is set, the SFN modulator will wait for an *enable* command before applying this function. See the *MIPFuncEnable* command.
If the *MIPFuncCellID* command is given with the *TXId* parameter only, the CellID Function for this TXId, if present, is removed and no longer present in the MIP.

6.4. MIPFuncFreqOffset

Used for: Add/Modify/Delete a Frequency Offset Function.
Parameters: TXId, FrequencyOffset

Example: MIPFuncFreqOffset 77,-12↵ *(Start transmitting a frequency offset of -12Hz to the transmitter with an ID of 77).*

Notes: If the *MIPFuncCellID* command is given with the *TXId* parameter only, the Frequency Offset Function for this TXId, if present, is removed and no longer present in the MIP.

6.5. MIPFuncPower

Used for: Add/Modify/Delete a Power Function.

Parameters: TXId, Power

Example: MIPFuncPower 77,70↵ *(Start transmitting a power function of 7.0dB to the transmitter with an ID of 77).*

Notes: If the *MIPFuncPower* command is given with the *TXId* parameter only, the Power Function for this TXId, if present, is removed and no longer present in the MIP.

6.6. MIPFuncPrivate

Used for: Add/Modify/Delete an Private Function.

Parameters: TXId,Data[,Data,Data]

Example: MIPFuncPrivate 77,0x00,0x33↵ *(Start transmitting a private function with a (0x00,0x33) payload to the transmitter with an ID of 77).*

Notes: If the *MIPFuncPrivate* command is given with the *TXId* parameter only, the Private function for this TXId, if present, is removed and no longer present in the MIP.

6.7. MIPFuncTimeOffset

Used for: Add/Modify/Delete a Time Offset Function.

Parameters: TXId,TimeOffset

Example: MIPFuncTimeOffset 77,50↵ *(Start transmitting a time offset of 5.0uS to the transmitter with an ID of 77).*

Notes: If the *MIPFuncTimeOffset* command is given with the *TXId* parameter only, the Time Offset function for this TXId, if present, is removed and no longer present in the MIP.
The TimeOffset parameter is given with a resolution of 0.1µs (100ns).

6.8. MIPFuncList

Used for: Show currently broadcast MIP functions.

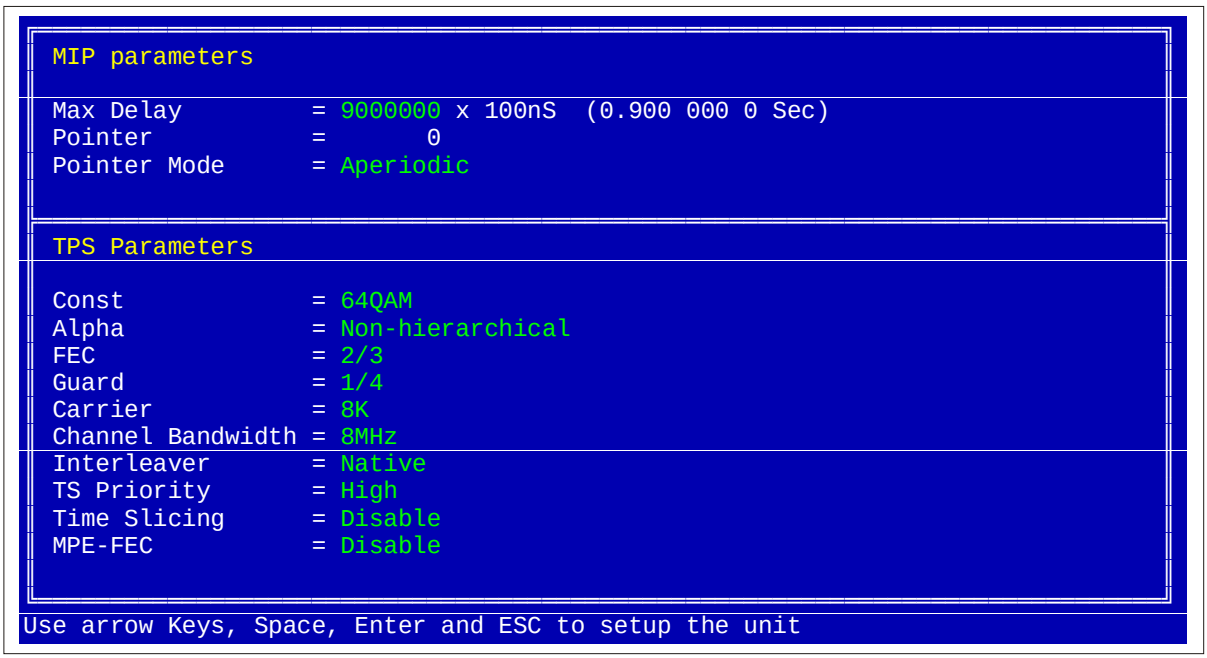
Parameters: None

Example: MIPFuncList↵

7. Interactive configuration and status commands

7.1. EasyConfig

Used for: Interactive MIP parameters configuration.
Parameters: None
Example: EasyConfig ↵



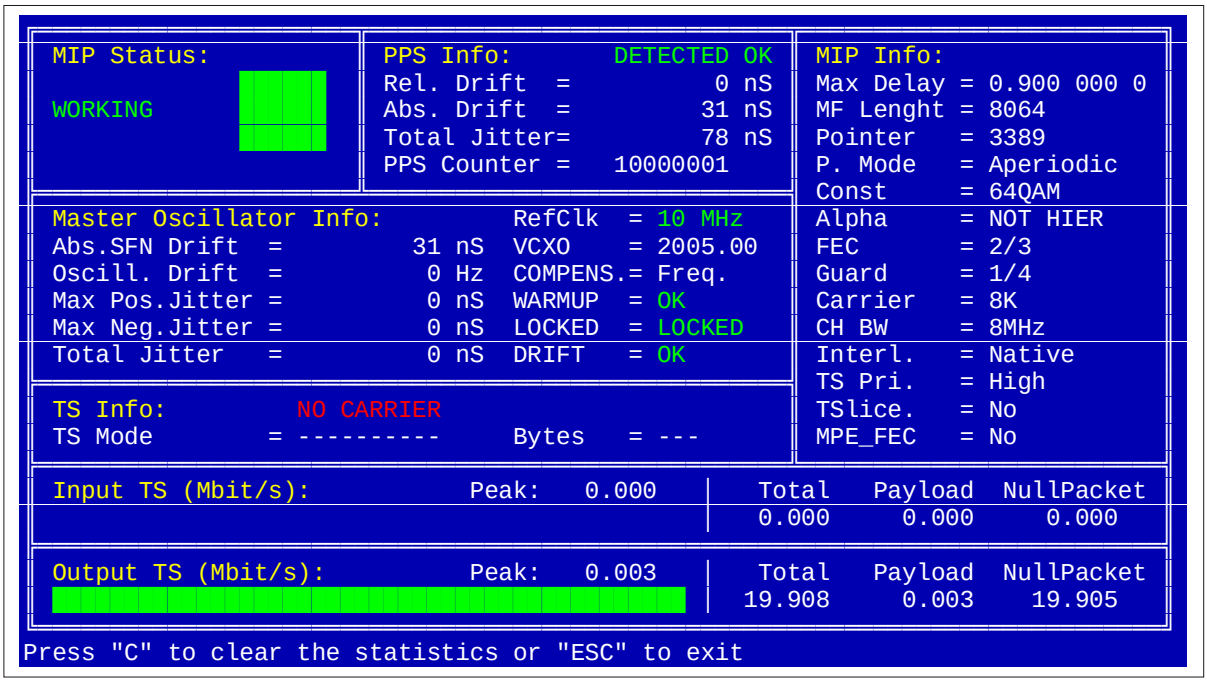
EasyConfig page example

7.2. TsMon

Used for: Show transport streams bitrates.
Parameters: None or 1 or 2
Example: TsMon ↵ *(Simple text output)*
TsMon 1↵ *(Numeric only –microcontroller– output)*
TsMon 2↵ *(ANSI continuous monitoring)*
Notes: The *EasyView* command is the same as *TsMon 2*.

7.3. EasyView

Used for: Shows all functional system information.
Parameters: TIME
Example: EasyConfig ↵ *(Default refresh delay of 0.1s)*
EasyConfig 10 ↵ *(Refresh every 1 second)*
Notes: Refresh rate is in 0.1s resolution.



EasyView page example

7.4. MIPStatus

Used for: Show MIP inserter status monitor.
Parameters: None or 1 or 2
Example: MIPStatus ↵ *(Simple text output)*
MIPStatus 1↵ *(Numeric only –microcontroller– output)*
MIPStatus 2↵ *(ANSI continuous monitoring)*
Notes: The *EasyView* command is the same as *MIPStatus 2*.
Returned values are MIP, PPS and VCXO (local oscillator) status.
For the numeric only output, values are as following:
MIP Status: 1=OK, 2=Error, 4=Starting, 8=OK but without PPS
PPS Status: 0=PPS OK, 1=PPS Missing
VCXO Status: Bit 0=Warmup OK, Bit 1=Locked, Bit 2=Drift error, Bit 3=Warmup Reset.

7.5. EasyFunctions

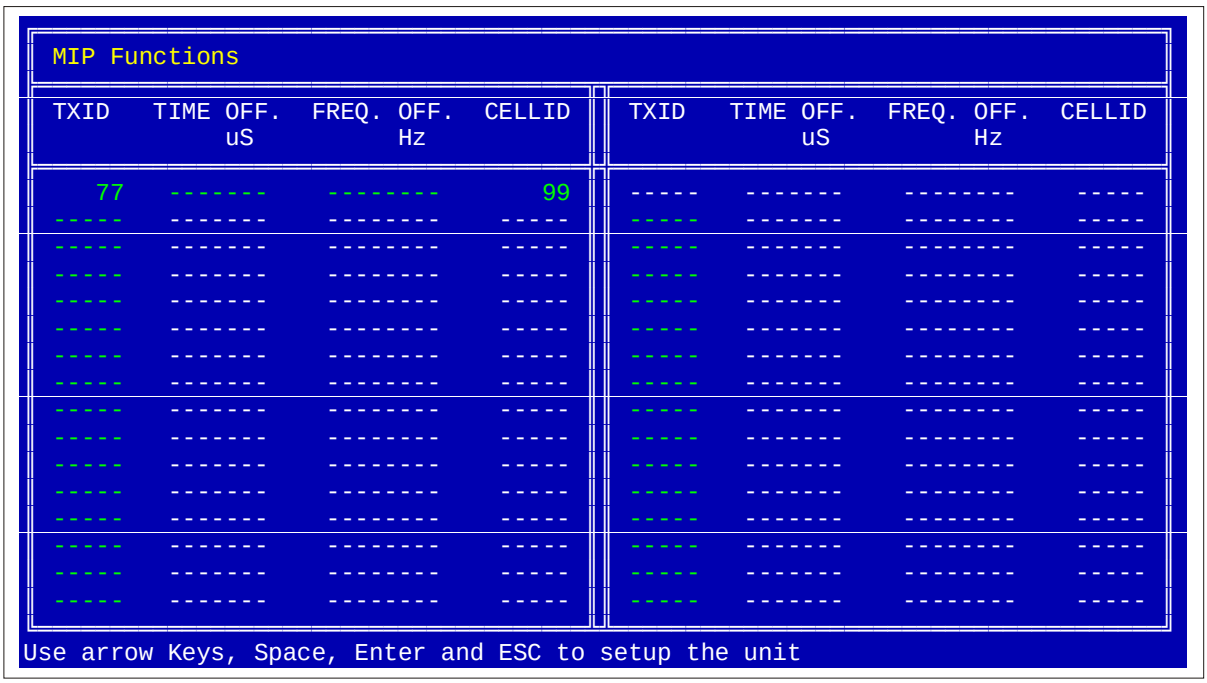
Used for: Interactive MIP function configuration screen.

Parameters: 0 or 1

Example: EasyFunctions 0↵ *(Changes are applied as they are entered)*

EasyFunctions 1↵ *(All changes are applied at the same time when exiting)*

Notes: The EasyFunctions command ease the configuration of three frequently used functions (Time offset, frequency offset and Cell ID) for several transmitters.



EasyFunctions page example

8. Options, transport stream and I/O configuration

8.1. SelASIInput

Used for:	Select or get ASI input.	
Parameters:	0,1	
Example:	SelASIInput 0↵	*(Use ASI TS input #1)*
	SetASIInput 1↵	*(Use ASI TS input #2)*
Notes:	It cannot be used if the ASI input is used as a reference clock input (see *RefClock* on page 21).	

8.2. SetASIInput

Used for:	Select normal or inverted ASI TS input.	
Parameters:	MODE	
Example:	SetASIInput 0↵	*(ASI TS input disabled)*
	SetASIInput 1↵	*(Select normal ASI TS input)*
	SetASIInput 2↵	*(Select inverted ASI TS input)*
Notes:	Inverted ASI inverts LVDS+ and LVDS- signals.	
	MODE: 0,1,2 (0 = Disabled, 1 = Normal, 2 = Inverted)	
	Both ASI Inputs have the same setup.	

8.3. SelASIOutput

Used for:	Set or get ASI output source.	
Parameters:	OUT	
Example:	SelASIInput 0↵	*(ASI output disabled)*
	SetASIInput 1↵	*(ASI output outputs SFN stream)*
	SetASIInput 2↵	*(ASI output outputs ASI input -bypass-)*
Notes:	Parameters are as follow:	
	OUT: 0,1,2 (0=Disabled, 1=MIP inserter output. 2=TS input bypass).	

8.4. SetASIOutput

Used for:	Select normal or inverted ASI TS output.	
Parameters:	MODE	
Example:	SetASIOutput 0↵	*(ASI TS output disabled)*
	SetASIOutput 1↵	*(Select ASI TS output)*
	SetASIOutput 2↵	*(Select inverted ASI TS output)*
Notes:	Inverted ASI inverts LVDS+ and LVDS- signals.	
	MODE: 0,1,2 (0 = Disabled, 1 = Normal, 2 = Inverted).	

8.5. LedSPI

Used for:	Set or get configuration of LED SPI.	
Parameters:	MODE	
Example:	LedSPI 0↵	*(LED SPI disabled)*
	LedSPI 1↵	*(LED SPI enabled, signalling polarity is positive)*
	LedSPI 1↵	*(LED SPI enabled, signalling polarity is negative)*

Notes: For the meaning of the serial SPI flags please refer to *Flags SPI sample schematic* on page 27.

8.6. RefClock

Used for: Set or get reference clock source.
Parameters: REF
Example:
RefClock 0↵ *(Derive 10MHz reference from PPS)*
RefClock 1↵ *(10MHz is fed to the dedicated pin)*
RefClock 2↵ *(10Mhz is fed to the differential ASI#1 input)*
RefClock 3↵ *(10Mhz is fed to the differential ASI#2 input)*
Notes: *RefClock 0* does not actually use a 10MHz external reference: it derives the 10MHz by frequency locking an internal clock to the PPS Hz. This option has an higher jitter compared to feeding a real 10MHz to the module.
If an ASI input is used as a differential 10MHz input it cannot be used as an ASI data input.

8.7. PPsEdge

Used for: Set or get GPS PPS edge polarity.
Parameters: 0 or 1
Example:
PPsEdge 0↵ *(Falling edge)*
PPsEdge 1↵ *(Rising edge)*

8.8. FlagsMatrix

Used for: Set or get leds configuration.
Parameters: A,B,C,D,E,F,G,H
Example:
FlagsMatrix 0x4210DE37↵ *(Default value for STK V1.3)*
FlagsMatrix 0x52314670↵ *(Default value for STK V1.4)*
Notes: Each letter represent a function (see table below) and is a number 0 to 15, which indicates the Flag output pin to route the function output to (0 to 7, add 8 for inverted output).
Different flags can be output to the same pin, if needed.

A	Overflow	E	MIP Error
B	Burst	F	SFN Unlock
C	188/204	G	OverTemp
D	Carrier Detect	H	SELFTest OK

The A,B,C,D,E,F,G,H can also be replaced by the hexadecimal number 0xABCDEFGH or the equivalent decimal value (so *4,2,1,0,D,E,3,7*, *0x4210DE37* and 1108401719 are equivalent).

8.9. MIPConfigOptions

Used for: Set or get MIP inserter configurations options.
Parameters: SWITCH,OUT,TRIG,LOOP
Example: MIPConfigOptions 1,1,0,0↵ *(Default options)*
Notes: Parameters are as follow:
SWITCH 1=Enable,0=Disable seamless options switch.

OUT 1=Enable, 0=Disable ASI output when MIP is not working
TRIG 1=Enable, 0=Disable STS counter resync at every PPS
LOOP 1=Enable, 0=Disable PPS and 10MHz loop to output pins.

Notes: *Seamless options switch* means that the ASI output stream is not interrupted when changing options, but the option change is delayed to be done at the beginning of a megaframe.
If TRIG is enabled, the internal counter is resynchronized at every PPS, instead of a single synchronization done when starting the MIP insertion. The single synchronization option is normally preferable because, provided that the 10MHz is a stable reference, the jitter present at the PPS input is not propagated.

8.10. MIPStartUpDelay

Used for: Set or get MIP insertion startup delay.
Parameters: DELAY
Example: MIPStartUpDelay 20↵ *(Set a startup delay of 2.0 seconds)*
Notes: Delay is given in 0.1s.
This startup delay is the time the module waits, after having achieved lock of the input TS, the 10MHz reference clock and the PPS signal, before starting TS output.

9. FLEXMod module upgrade

IMPORTANT NOTE:

If the module has the "**FLEXMod 2.0 FPGA Boot Loader**" installed, do not use the following procedure to upgrade.
Please refer to the "**FLEXMod 2.0 FPGA Boot Loader**" documentation for a fail-safe and faster upgrade procedure.

The FLEXMod firmware can be user upgraded if needed.
The upgrade procedure is performed with the following steps:

1. Change the baud rate to something faster than 115200. This step is not necessary (the upgrade can also be done at any baud rate) but since the firmware is quite long it is advisable to do this.
2. Issue the *FlashFormat 741852* command which erases the onboard serial flash containing the firmware.
3. Wait for the FLEXMod signaling the end of the flash erasing procedure (it takes normally 10 seconds)
4. Issue the *Upgrade 741852* command and upload the new firmware using the XMODEM protocol.
5. When the upload has been completed, wait for the acknowledge that the upgrade has been successfully completed.
6. Power off and then back on the FLEXMod.

Please note:

Should the procedure fail for any reason, perform the procedure again starting from step 2. **Do not** power off the board since the flash has been erased (or contains an invalid firmware) and hence the board will not work (and could also, in rare occasions, be damaged). Should this happen, the board has to be returned to factory for reprogramming

9.1. FlashFormat

Used for:	Erases the onboard serial flash.
Parameters:	741852
Example:	FlashFormat ↵
Notes:	The *741852* parameter is a constant value used to avoid the risk of issuing this command by chance. Do not erase the flash if you are not ready to perform an upgrade procedure. After the flash has been erased, the FLEXMod will not be able to boot.

9.2. Upgrade

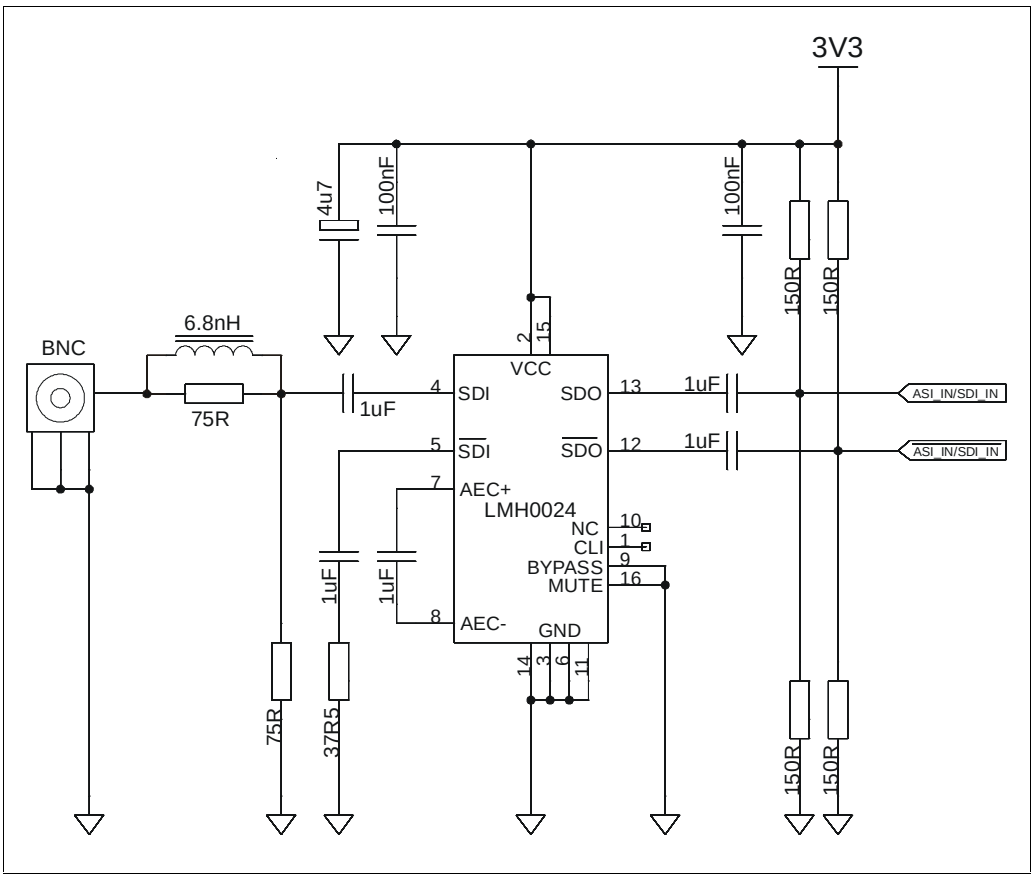
Used for:	XMODEM upload of a new firmware.
Parameters:	741852

Example: Upload↵
Notes: Any terminal software capable of the XMODEM protocol can be used.
The *741852* parameter is a constant value used to avoid the risk of issuing this command by chance.

10. Sample schematics

10.1. ASI Input sample schematic

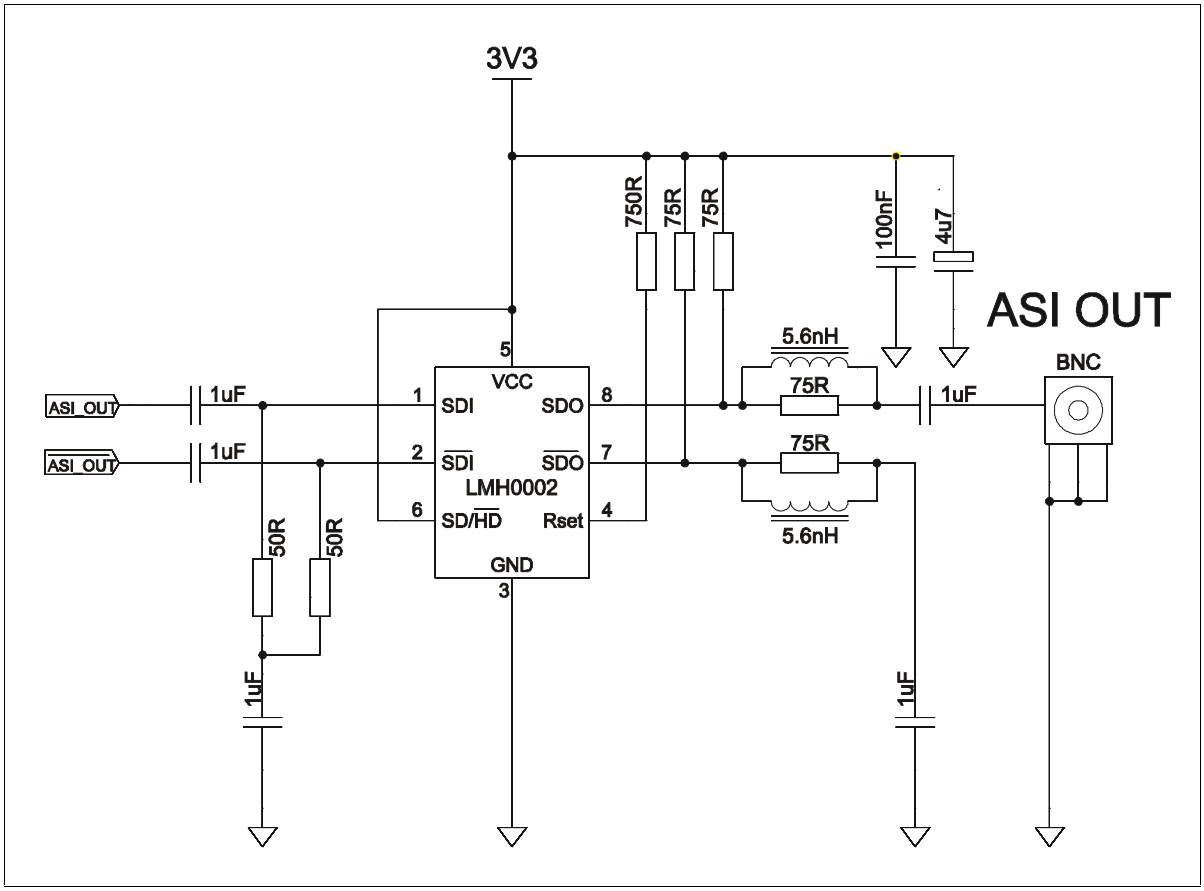
The LDVS TS ASI Input requires a cable equalizer and an adaptation network in order to adjust input levels. Note that ASI_IN is a LVDS signal, so route accordingly. The following schematics is a typical application ASI input:



Typical ASI input

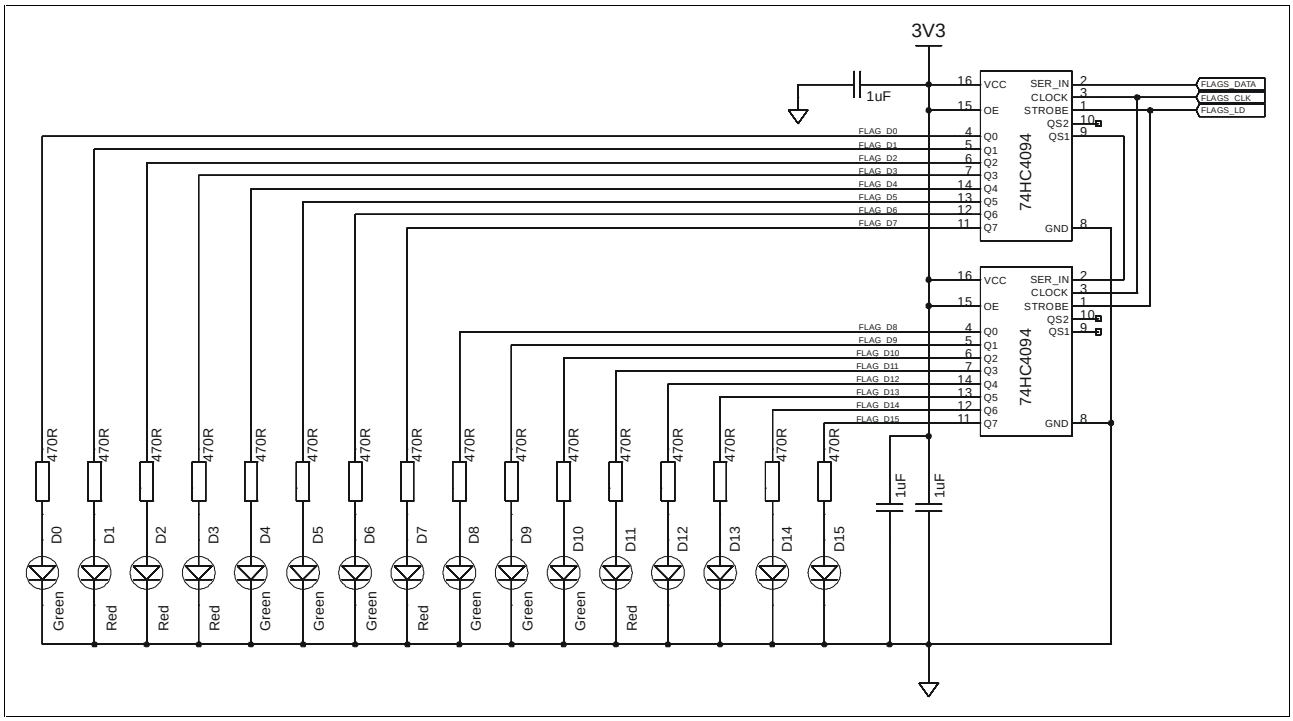
10.2. ASI Output sample schematic

The LDVS TS ASI Output requires an adaptation network in order to adjust input levels and an output driver. Note that ASI_OUT is a LVDS signal, so route accordingly. The following schematics is a typical application ASI output:



Typical ASI output

10.3. Flags SPI sample schematic



Bit	Flag
D0	Boot OK
D1	OverTemp
D2	SFN Unlock
D3	MIP Error
D4	Selected ASI TS DCD
D5	Selected ASI TS 204/188
D6	Selected ASI TS Burst
D7	Selected ASI TS Overflow

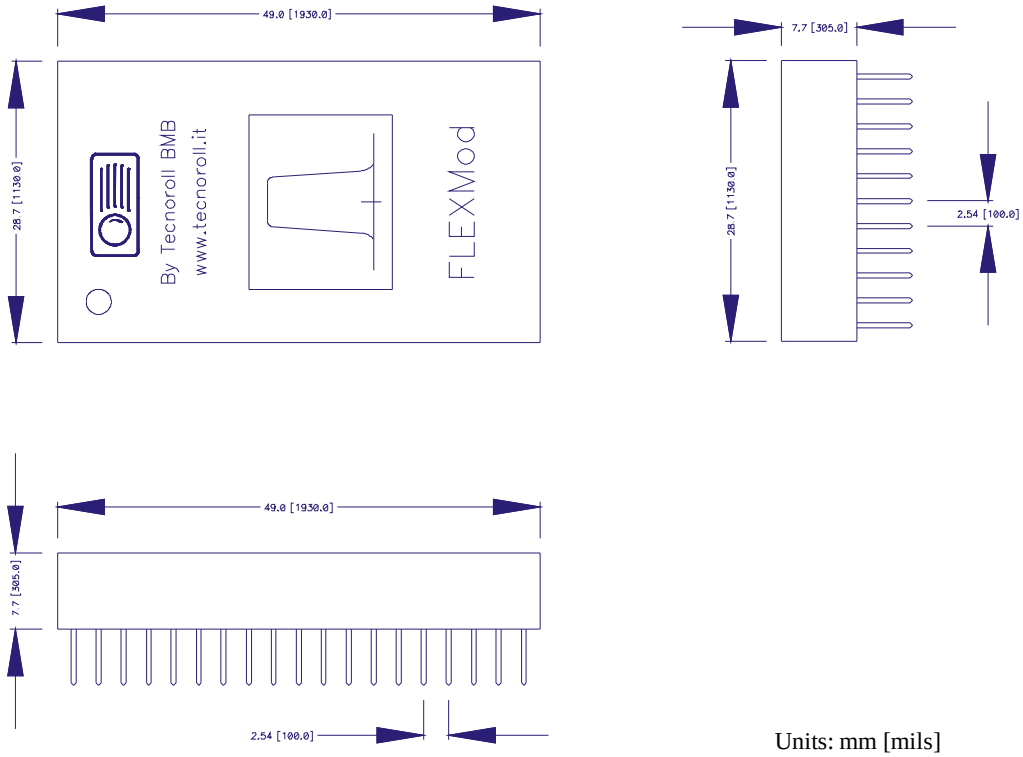
Bit	Flag
D8	Spare / Unused
D9	Spare / Unused
D10	Spare / Unused
D11	Spare / Unused
D12	Power ON
D13	Spare / Unused
D14	Spare / Unused
D15	Spare / Unused

11. Power supplies characteristics

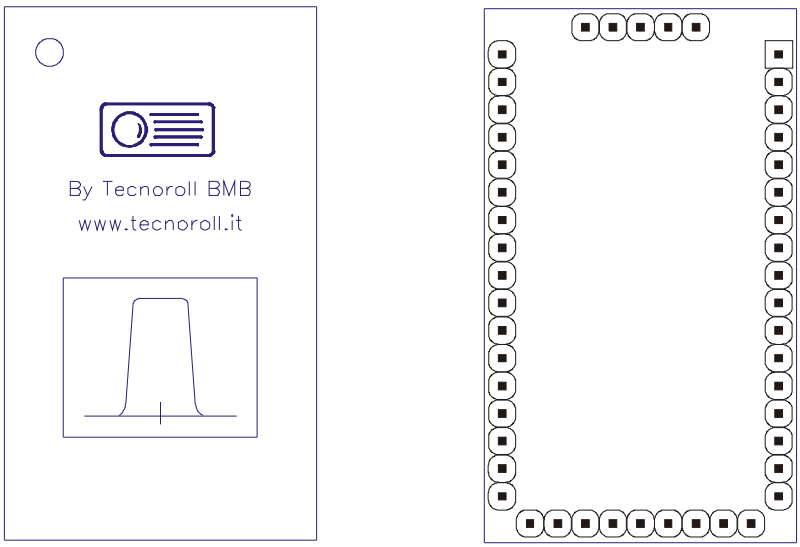
SYMBOL	PARAMETER	CONDITIONS	MIN	TYP.	MAX	UNIT
1V2	Core voltage			TBD		mA
1V8	DAC voltage			TBD		mA
3V3	I/O voltage			TBD		mA

12. Alphabetical command list

13. Packaging information



Units: mm [mils]
Controlling dimensions: mm

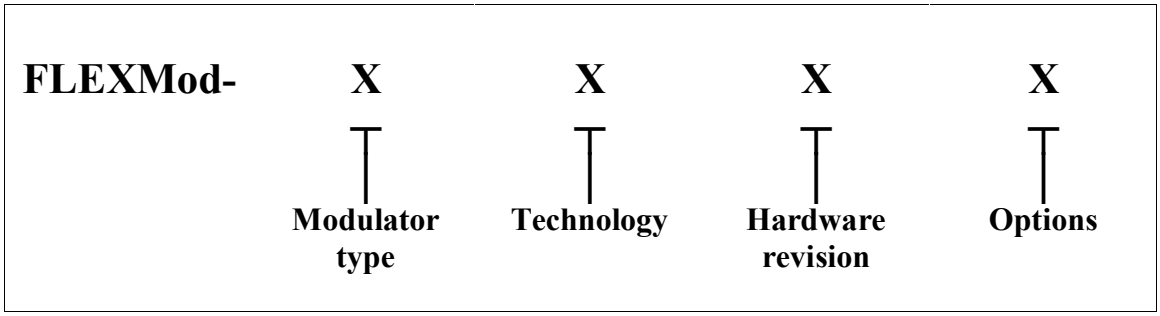


Top View Bottom View

Appendix A 'Type' description for connector pins

Type	Description
I_a	Analog input
I_5	5V compatible input
I_3	3.3V compatible input
$I_{3/5}$	3.3V compatible input, 5V tolerant
$I_{LVDS\pm}$	Positive/Negative LVDS input
O_a	Analog output
O_3	3.3V output
O_5	5V output
OC_3	3.3V output, Open Collector
$O_{LVDS\pm}$	Positive/Negative LVDS output
P	Power supply line
P_o	Power supply output line

Appendix B FLEXMod Product Identification System



Modulator type	
0	Unprogrammed hardware
1	DVB-T MFN: *ETSI EN 300 744* compliant COFDM modulator.
2	DVB-S: *ETSI EN 300 421* compliant QPSK modulator.
3	DVB-C: *ETSI EN 300 429* compliant QAM modulator.
4	ATSC (A/53) compliant modulator.
5	DVB-S/S2: *ETSI EN 300 421/302 307* compliant QPSK/8PSK/16,32APSK modulator.
6	DVB-T SFN: *ETSI EN 300 744/TS 101 191* compliant COFDM modulator.
7	SFN Adapter/MIP Inserter
8	Aspect Ratio corrector and EPG inserter
9	Transport Stream Converter/Remultiplexer
AV1	DVB-T MFN: ETSI EN 300 744 compliant COFDM TR391AV Controller
AV2	DVB-S: ETSI EN 300 421 compliant QPSK TR391AV Controller
AV3	DVB-C: ETSI EN 300 429 compliant QAM TR391AV Controller
CM0	Transport stream Combiner

Technology	
0	Reference clock: 1.024 GHz.
1	Professional tunable reference clock (VCXO): 1.024 GHz.
2	Professional tunable reference clock (VCXO): 1.024 GHz and SFN input FIFO
8	Reference clock: 16 MHz.

Options	
P	Premium type.
N	Unboxed (naked) version.
NP	Unboxed (naked) version, no pins, panelized

Examples:
FLEXMod-101 Consumer grade DVB-T modulator, reference clock 1.024GHz, HW revision 1
FLEXMod-281 Consumer grade DVB-S modulator, reference clock 16MHz, HW revision 1