Dokumentenart/ *Type of Document*	Klassifizierung/ *Classific.*	Ablageinformation (Reg.-Nr./Dateiname)/ *Inform. for filing*
Technical Requirement Specification		

Von/*from:*

Ort/ *Pl.*	Abt./*Dept.*	Name/*Name*	Tel. int./ *Phone*	Fax./*Fax*	Datum/ *Date*	Seit./*Pages*
Ba	**Design**	**Meng Engineering M.Meng**	**+41 56 222 44 10** **+41 79 230 93 86**	**+41 56 222 44 10**	**22.01.00**	**1 of 30**

Betrifft/ *concerns:*

HW-TRS : Fieldbus Interface Coupler KP99 Rev. A

1. Purpose

The description of the requirements and the design constraints taken for the design of the AF100 fieldbus interface coupler **KP99** for the SPS KT94.

2. Summary

The **KP99** interface coupler shall provide all the necessary interface logic in order to connect the KT94 SPS with the AF100 fieldbus. The connection must be made through a simple, standard 40 poles double row connector on a flat ribbon cable. The **KP99** coupler has its own power supply. All the data exchange between the KT94 and the **KP99** is done through a simple - 80188 intel like - 8-bit wide and 16KB deep memory mapped dual ported interface. Besides some logic in a FPGA and the AF100 bus coupler ASIC MVBC [MVBC], no additional processor is needed on the interface card. The SW for the configuration and use of the fieldbus interface runs on the KT94 SPS. The interface card on its side shall provide the following functionality:

- Class 2 or higher Device Support [MVBC].
- Hardware and software configurable Device Address. 1.5 Mbit/s data rate.
- Support for redundant RS485 tranceiver interface.
- Support for redundant optical interface using fibre optical transmitter and receiver components.
- 256KB of SRAM based traffic memory [MVBC].
- KT94 compatible - intel 80188 like - dual ported memory mapped hardware interface. The complete MVBC traffic store must be made visible through a configurable address windowing mechanism.
- Standard 24 volt DC power supply input.
- Robust module case. It must be insensitive to EMI. Simple DIN compatible module mounting mechanism.

2.1. Restrictions

- The fibre optical components used are optimized for the standard 62.5/125 um telecom glass fibre.

A revision history can be found on the last page of the document.

Visum / *Signature*

TABLE OF CONTENTS

LIST OF FIGURES

3. Scope

This document summarizes the requirements for the hardware design of the AF100 fieldbus interface coupler **KP99**. The programmers interface - on a register and memory map level - from the software side is specified as well. However no KT94 specific information - except for the DPM interface pin and signal description - is found in this document. Information concerning the AF100/MVB fieldbus coupler ASIC MVBC can be found in [MVBC].

3.1. Abbreviations and Definitions

3.1.1. Abbreviations

AF100	Advant Fieldbus 100.
ASIC	Application Specific Integrated Circuit.
CPLD	Complex Programmable Logic Device.
DA	Device Address [MVB][MVBC]
DA-PIT	Port Pointers for the Device Address Ports [MVBC]. For DA reception only.
DIL	Dual in Line (Package, Socket).
ESD	Electro Static Discharge.
ESD	Electrical Short Distance Bus. In the sense of AF100/MVB bus topologies.
FPGA	Field Programmable Gate Array.
GND	The digital ground.
HW	Hardware.
ISP	In-Circuit Programmable.
KB	Kilo Byte.
KP99	This abbreviation is used as a name for this AF100 interface module.
LA-PIT	Logical Address for MVB Ports [MVBC].
MCM	Memory Configuration Mode [MVBC]. Defines the size of the traffic store.
MVBC	Multifunction Vehicle Bus Controller.
PCB	Printed Circuit Board.
PCS	MVB Port Control and Status register [MVBC]. Part of the traffic memory.
PIT	MVB Port Index Table [MVBC]. Part of the traffic memory.
SMD	Surface Mount Device.
SMT	Surface Mount Technology.
SPS	Speicher Programmierbare Steuerung.
SW	Software.
TS	Traffic Store.
VDD	The digital +5 volt supply.

3.1.2. Symbols

A vertical bar seen on the left/right side of the document indicates a difference in the contents compared to the previous revision. Minor changes (i.e. spelling) are not marked with this bar.

3.2. Document Structure

- Chapter 4 provides general information about the AF100 fieldbus coupler module.
- Chapter 5 contains the programmer's model for the **KP99**.
- Chapter 6 specifies the operational condition as well as the handling for the module.

3.2.1. Definitions

3.2.1.1. Active State

A bar "/" at the beginning or and "N" or "n" at the end of a signal and/or pin name means that the active state is electrically "low".

If a signal is asserted, it is activated. This is independent of what the active level is.
If a signal is negated, it is not active. This is independent of what the active level is.

Positive logic means, that a signal is asserted when the logic level is high. The logic level is '1'.

Negative logic means, that a signal is asserted when the logic level is low. The logic level is '0'.

3.2.1.2. Bits and Byte Order

Long Word	Understood as a 32-bit wide word unless specified otherwise.
High Word	Understood as the 16-bit wide word from the range [31..16].
Low Word	Understood as the 16-bit wide word from the range [15..0].
Word	Understood as a 16-bit wide word.
High Byte	Bit15-8 of a word. Even Byte!
Low Byte	Bit7-0 of a word. Odd Byte!
High Nibble	Bit7-4 of a Byte.
Low Nibble	Bit3-0 of a Byte.
Bit Order	Bit31 = The most significant bit (MSB). The most left bit. Bit0 = The least significant bit (LSB). The most right bit.

The terminology *low/high byte* and *even/odd byte* is used in the sense of the MOTOROLA Big Endian format! When, for some reasons the INTEL Little Endian format is meant it is stated explicitly!

Big Endian Format (MOTOROLA)
- High byte D[15:8] (even byte) is stored at the address location *i*.
- Low byte D[7:0] (odd byte) is stored at the address location *i+1*.

Little Endian Format (INTEL)
- Low byte D[7:0] (even byte) is stored at the address location *i*.
- High byte D[15:8] (odd byte) is stored at the address location *i+1*.

3.2.2. Stylistic

An example of a register description is given below:
Generic Register: (This register is not part of the KP99. It is used as an example only!)

Generic Register (GR) D[15:0]: Address(byte count) 0x0'FFAE

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	ABC	EFG	P12	P11	P10
0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1
-	-	-	-	-	-	-	-	-	-	-	R0	RW	RW	R	R1W

1st Row: Bit assignment of the register. A lower-case "x" indicates an unused bit. This bit cannot be used for general-purpose data storage.

2nd Row: Indicates the initial value after reset. If an "x" is found on the 1st row, then the specified value in the 2nd row will always be read out. Reading values other than specified reflect a chip malfunction.

3rd Row: Indicates access rights for the local CPU (upper-case symbols):

-	Bit not existing or supported
R	Readable
R0	Read as zero all time
R1	Read as one all time
W	Writable

Combinations of the above are allowed.

3.3. References

3.3.1. ABB and ADtranz documentation

[MVB] IEC 61375-1999, Clause 3
[TCN_003] IEC61375 TCN Nr. 9 Electric Railway Equipment. Part 3.

[MVBC] MVBC Data Sheet. 3EHL420441.
[REGA] REGA ASIC Datasheet. Component from ADtranz.

[REF_01] KP 92 hardware schematics page 1 and page 5. Mai 1994
[REF_02] Technische Notiz. ABB Procontic CS31 AF100 Ankopplung für 07MR93. 28.05.1998
[REF_03] ADtranz reference design, MVB RS485 interface connection. Peter Sandberg 96-10-17, DRH96061ALL.

3.3.2. Other documentation

[JTAG] IEEE 1149.1-1990. IEEE Standard Test Access Port and Boundary-Scan Architecture.
[VHDL] VHDL for Programmable Logic. ISBN 0-201-89586-2.

3.3.3. Web Resources

[ALTERA_001] FLEX 10K Embedded Programmable Logic Family. Data Sheet Version 3.13.
[ALTERA_002] Configuration EPROM's for FLEX devices. Data Sheet Version 9.
[ALTERA_003] ALTERA device package information. Data Sheet Version 7.01.

[CYP_001] Cypress Semiconductor. CY7C130/CY7C131 and CY7C140/CY7C141 1K x 8 Dual-Port Static Ram.

[HP_001] HP Components. Low Cost Miniature Fibre Optic Components HFBR-0400 series.

4. System Architecture - Function Block Decomposition

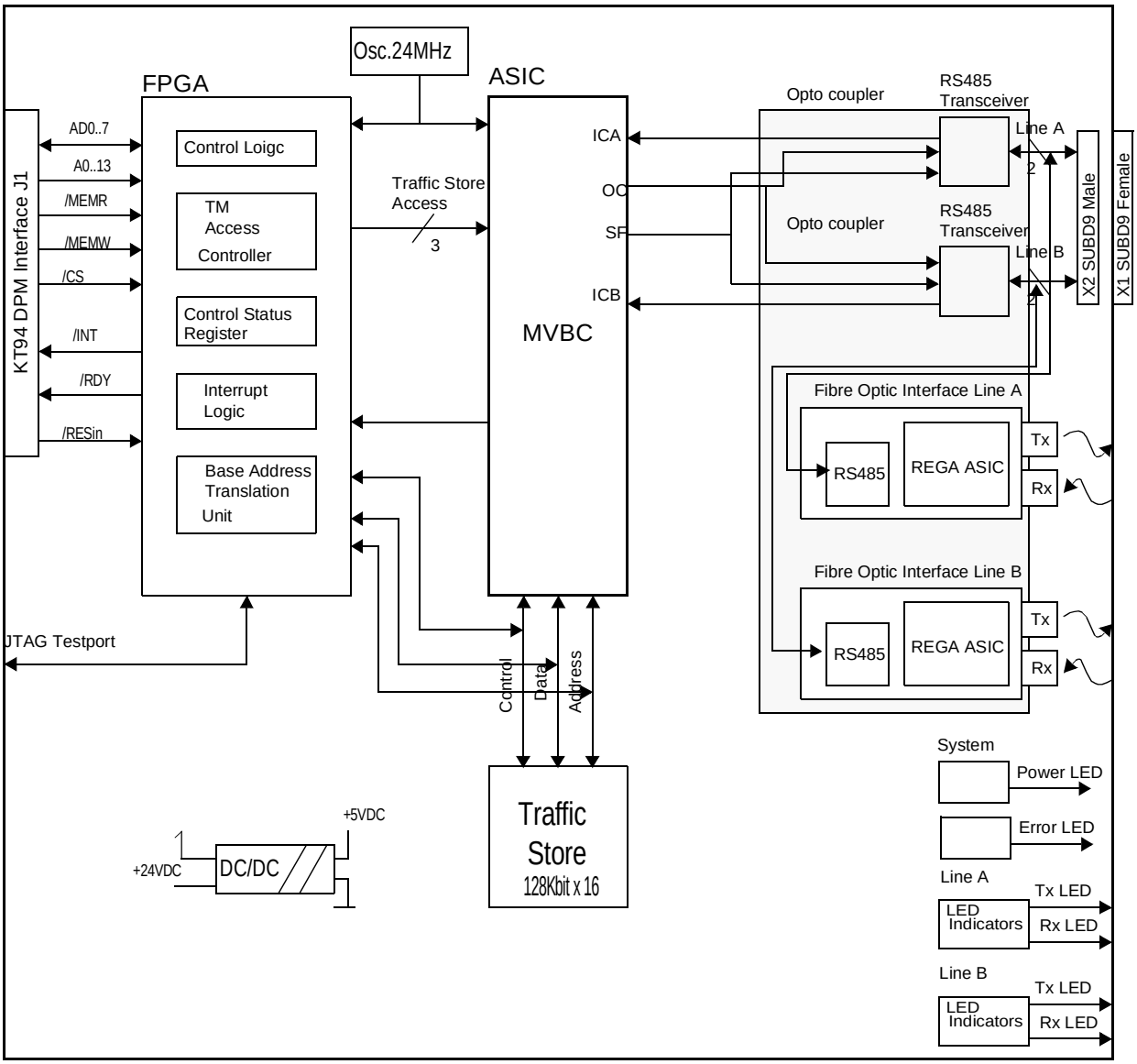


Figure 1: **KP99** Block Diagram

4.1. Overview

The **KP99** module shall be build around the MVBC ASIC [MVBC], the bus coupler chip for the AF100/ MVB fieldbus. The MVB subsystem contains the MVBC ASIC, 256KB of traffic memory for the shared data, and the 24MHz clock source. The FPGA shall provide the logic to interface **KP99** module with the KT94. It shall contain all the control and glue logic for the following functionality:

- Dedicated KT94 DPM interface.
- A traffic memory windowing mechanism. Through this mechanism it shall be possible to access the whole traffic memory and the MVBC internal registers in an effective way.
- A simplified interrupt logic.
- A bus request logic for MVBC internal register access as well as for the traffic memory.
- Additionally control and status and signalling logic.

The subsystem on the right side shall contain the RS485 driver section including the protection elements for the differential voltage spikes exceeding the RS485 specification. Additionally a redundant

optical interface for the AF100/MVB fieldbus must be provided. The RS485 bus interface and the optical interface shall be galvanic isolated from the main electronic side. It is therefore powered through the 24V DC/DC converter. Isolation withstand voltage shall be more or equal to 500V DC.

Some information of the activity on the AF100 fieldbus shall be provided through transmission and reception indicators using LED's. One status LED for the complete **KP99** shall act as a power indicator, and one LED for the complete **KP99** shall act as an error indicator. The error LED shall be set through software on the KT94 side.

4.1.1. Interrupt Support

One interrupt request line INTn is available on the KT94 DPM interface. It shall be possible to enable this interrupt line towards the KT94. However per default the interrupt line shall be disabled. Configuration inside the FPGA shall enable, that the two interrupt output sources from the MVBC ASIC are being routed to the INTn line of the KT94 DPM interface.

4.1.2. AF100/MVB Class Mode Support

The **KP99** supports MVB Class 2 and higher devices. The class1 mode, where no additional software is required for communication via the MVB is **not** supported.

4.1.3. AF100/MVB Device Address

The needed device address shall be set through software using the MVBC internal device address registers [MVBC].

4.1.4. AF100/MVB Traffic Store Access

The traffic store is 128Kbit x 16 in size. It is accessible as a **word port** only [16b
it]. Reading a byte or a word is the same except for some byte swapping needed depending on the value of the lowest address line driving the DPM interface. Writing can only be done as a word write. Therefore the first byte write is latched and hold back until the second byte write is done from the KT94 side. Then the word is written into the MVBC's registers or into the traffic store. A write operation **must** always be done in the same way i.e. in the same order. If not, no write operation is performed at all!

4.1.4.1. Traffic Store Write Operation

- Write the first byte to the even address port A_i.
- Write the second byte to the odd address port A_{i+1}.

After the second write the word is being written to the traffic store. Byte alignment inside the 16Bit word will always be treated in the following way:

- The first byte to the even address A_i is mapped to the data D[15:8] of the word inside the traffic store.
- The second byte to the odd address A_{i+1} is mapped to the data D[7:0] of the word inside the traffic store.

4.1.4.2. Traffic Store Read Operation

- Reading the byte at the even address A_i will **always** return D[15:8] of the traffic store.
- Reading the byte at the odd address A_{i+1} will **always** return D[7:0] of the traffic store.

4.1.5. AF100/MVB Connection

This subsystem on the right side shall contain the opto-couplers, the RS485 drivers and the optical interfaces. Both lines - line A and line B - shall be supported. This subsystem shall have its own galvanic isolated power supply. However additionally zero ohm resistors shall be provided on the **KP99** module in order to connect this subsystem with the main logic parts directly.

4.1.5.1. RS485 ESD Bus Connection

External connection for the RS485 bus lines shall be done using two SUBD9 DIN 41652 connectors

with M3 female screwlocks. The two connectors shall have identical pin assignments, but opposite pin directions (male and female) for easy network connection. X2 shall be a male and X1 shall be a female connector. The opposite pin direction shall allow:

- One type of AF100/MVB cable.
- Connecting two AF100/MVB cables together when a node is being removed.

The signals for the RS485 bus interface shall be connected in the following way. The trace length between X1/X2 shall not exceed 5 cm.

Table 1: X1/X2 Signal Assignment and Pin Description

Signal Name	Direction/Type	Description	Connector, Pin
A.Data_P	bidirectional	AF100/MVB Line A+ from RS485 line A signal pair.	X1: 1 X2: 1
A.Data_N	bidirectional	AF100/MVB Line A- from RS485 line A signal pair.	X1: 2 X2: 2
TxE	optional	Transmission/Reception Control Signal No connection shall be made.	X1: 3 X2: 3
B.Data_P	bidirectional	AF100/MVB Line B+ from RS485 line B signal pair.	X1: 4 X2: 4
B.Data_N	bidirectional	AF100/MVB Line B- from RS485 line B signal pair.	X1: 5 X2: 5
A.Bus_GND	supply	For termination resistance	X1: 6 X2: 6
B.Bus_GND	supply	For termination resistance	X1: 7 X2: 7
A.Bus_5V	supply	For termination resistance	X1: 8 X2: 8
B.Bus_5V	supply	For termination resistance	X1: 9 X2: 9

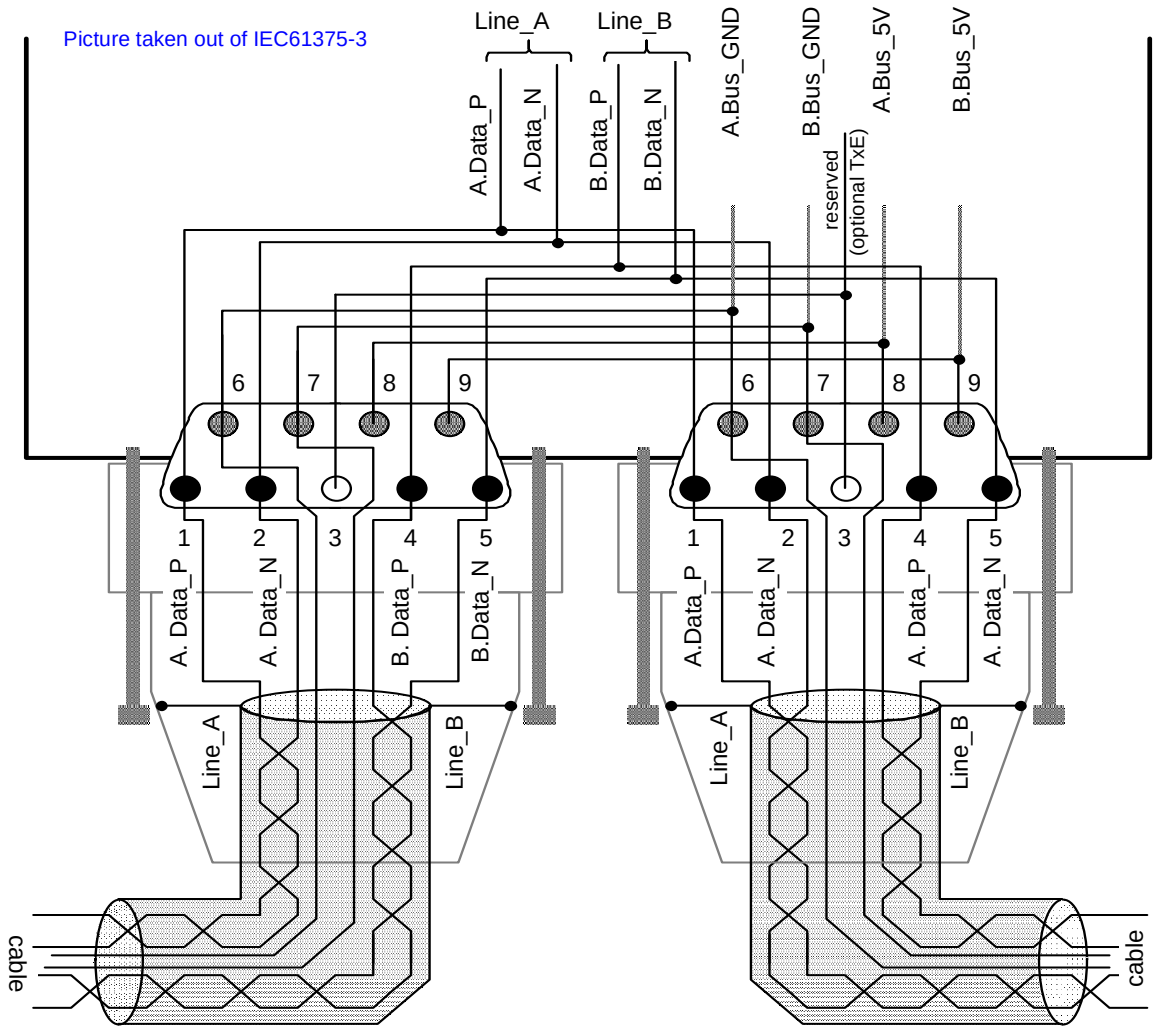


Figure 2: Recommended ESD Connection Diagram.

4.1.5.2. **KP99** Device Interconnection on the RS485 Bus

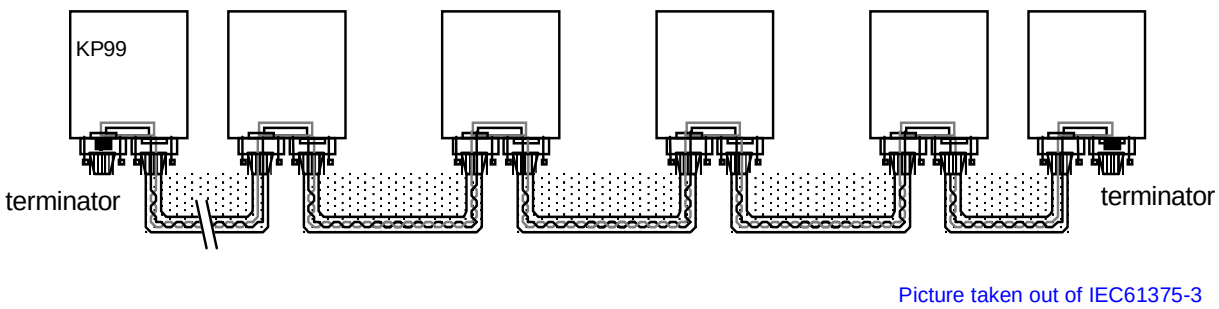
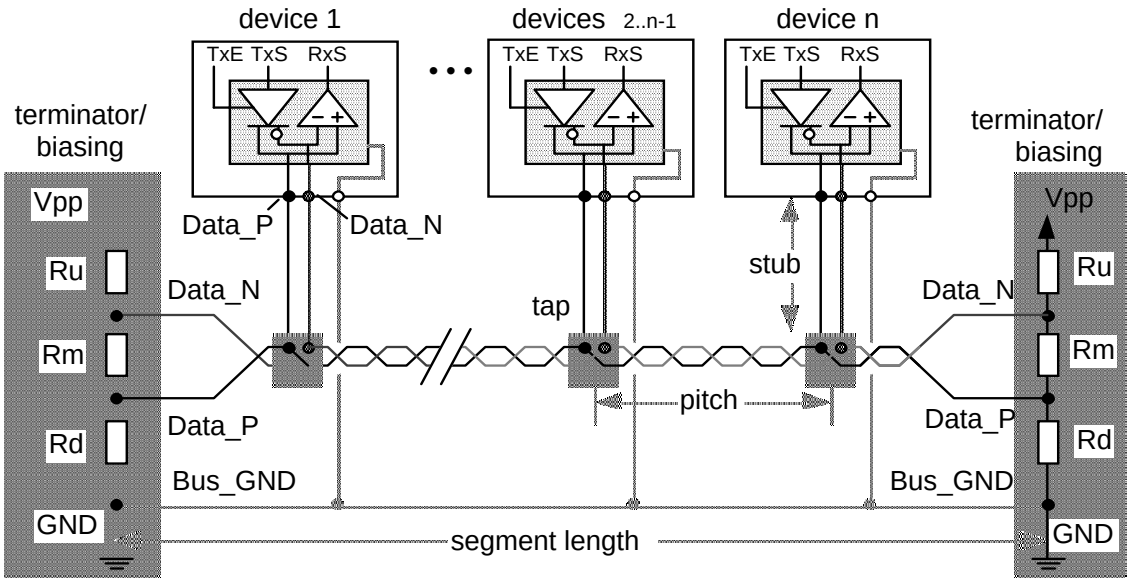


Figure 3: RS485 KP99 Device Interconnection

4.1.5.3. MVB RS485 Line Termination

On the **KP99** board only a weak pull-up and pull-down network is used. Mainly to guarantee a clean signal from the receiver section inside the RS485 transceiver. The ~100 Ohm passive termination must be done on both ends of the RS485 line. The termination network as shown below is **not** part of the **KP99** interface module! The figure below shows the needed termination network.



Components	Type	Value	Connections	Type	Value
Ru	resistor	390 Ω	Vpp	supply voltage	5,0 V
Rm	resistor	150 Ω		reference voltage	0,0 V
Rd	resistor	390 Ω	Gnd		

Figure taken out of the TCN norm part 3

Figure 4: EMD topology showing the required termination network.

4.1.5.4. Optical Connection

The optical interface shall be optimized for 62.5/125 um glass fibre cable. A standard multimode gradient index type with ~ 3..4dB attenuation/km cable length shall be used. Appropriate opto devices from HP shall be used (ST-type bayonet connectors). The following parts are recommended:

- Transmitter: HFBR-1414 [HP_001]
- Receiver: HFBR-2412 [HP_001]

The optical interface shall be designed to accommodate for glass fibre optic cable length starting at 0.1 meter up to 1000 meter. The type of fibre optical connection shall be of type ST.

The optical connection shall have line redundancy as well. It shall be possible to use the optical interface as a repeater between an ESD bus segment and an optical star bus topology. In order to optimize cost, it shall be possible to use **one** optical interface for both lines. The repeater functionality requires a signal regeneration circuit. The field proven REGA ASIC has to be used for this functionality!

4.1.5.5. AF100/MVB Interface Options

For cost reasons it shall be possible to manufacture KP99 modules with the following options:

- No optical interface at all.
- One optical interface at all.

4.1.6. Clocking Network

The MVBC and the REGA ASIC do require a 24Mhz oscillator having +- 100ppm deviation over temperature and time.

4.1.7. Reset Strategy and Power-Up Behavior

The reset signal RESETn must be driven from the KT94. Additionally the MVBC subsystem on the **KP99** shall be resetable through a register inside the FPGA. On some old MVBC chips, it has been mandatory to release the MVBC reset signal in phase with the 24Mhz clock - rising edge. Power-Up reset may need an additional delay before the FPGA on the **KP99** is operational. Up to 500ms is allowed! This may be true for Power-Up RESET only! A warm reset shall not generate an additional delay before the FPGA is accessible.

4.1.8. Support Status LED's

4.1.8.1. AF100/MVB Tx/Rx Signal Indication

The Transmit and the Receive LED indication logic is connected directly to the MVBC. A signal change on pin OC [MVBC] will trigger a monoflop, which will turn on the transmit LED for ~100msec. When the MVB subsystem on the MVBIP is addressed more than once within ~100msec, then the transmit LED will be turned on constantly.

A signal change on pin ICA **or** ICB [MVBC] will trigger a monoflop, which will turn on the receive LED for ~100msec. When the MVB subsystem on the MVBIP is addressed more than once within ~100msec, then the receive LED will be turned on constantly.

4.1.8.2. Error Indication

The error LED can be used as follows:

- As long as the FPGA is not properly configured, the error LED is **on**.
- After proper configuration of the FPGA the error LED is turned **off**.

- If needed, a bit in the diagnostic register can be asserted in order to turn the error LED on again. This can be used to indicate malfunction through software.

4.2. KT94 DPM Connector J1 Signal Assignment

The 40 pin double row standard connector J1 for the DPM interface has the following signal assignment:

Table 2: J1 Signal Assignment

Description/Meaning	Pin Number
VCC. +5Volt supply pins driven from the KT94.	1, 2, 3, 4
GND. Board Ground.	19, 20, 29, 30
A[0:13]. DPM Address Lines A[0..13]. 16KB address range.	5, 6, 7, .., 15, 16, 17, 18 A[0] connects to Pin 5, A[1] connects to Pin 6, ..., A[13] connects to Pin 18
D[0:7]. DPM Data Lines D[0..7]. Byte Port.	21, 22, 23, ..., 27, 28 D[0] connects to Pin 21, D[1] connects to Pin 22, ..., D[7] connects to Pin 28
/CS. KT94 DPM Chip Select Line. Low Active.	31
/MEMR. KT94 DPM Read Line. Low Active.	32
/MEMW. KT94 DPM Write Line. Low Active.	33
/RDY. KT94 DPM Ready Line. Falling Edge is buscycle termination signal.	34
/INT. Interrupt Request Line driven from the AF100 interface card. Low Active and level sensitive interrupt.	35
/RESin. System Reset Signal. Driven from the KT94. Low Active.	36
Unknown. Not Connected!!	37, 38, 39, 40

5. Programmers View

5.1. MVBC Traffic Store

The complete traffic store and the internal registers of the MVBC are accessible from the KT94 DPM using an address windowing mechanism. There is a maximum of 1MB MVBC memory area [MVBC]. However the **KP99** shall be able to support 256KB of traffic memory.

For MVB class 2 devices or higher [MVBC][TCN_003], the MVBC memory configuration mode must be set to 3 MCM[2:0] = 3 [MVBC]. The supported data type is word - 16 bit only!!!

5.1.1. MVBC Start-up and Configuration (MCM[2:0] = 0)

Whenever the MVBC shall be used for class 2 or higher devices, it will start up first with the MVBC memory configuration mode zero - MCM[2:0] = 0. According the datasheet [MVBC] the memory map is the following:

Picture "copied" from [MVBC]

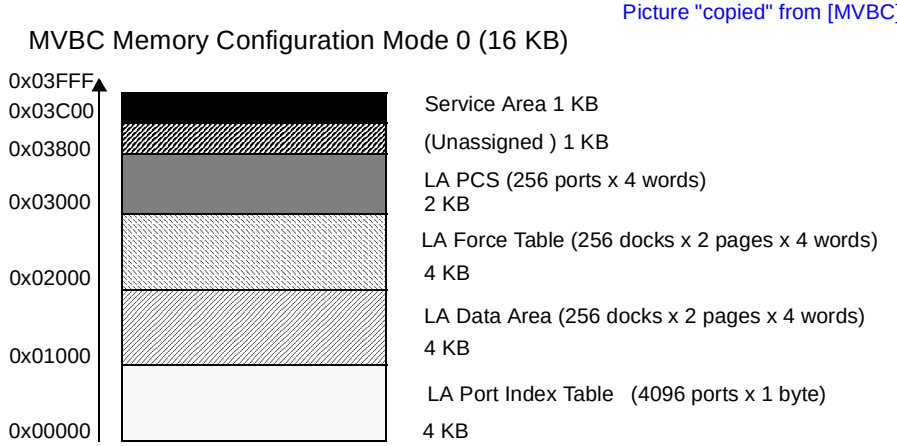


Figure 5: MVBC Start-up with MCM[2:0] = 0 for Class 2 or higher Devices [MVBC]

Now the first configuration accesses must be done in the service area of the MVBC [MVBC] in order to change to the required memory configuration mode three. Please note that the effective address for the MVBC service area does change as well [MVBC]. Therefore after changing the MCM[2:0] bits of the MVBC the MCM[2:0] bits of the BAR_CSR (page 23) must be changed as well!

5.1.2. MVBC Requested Memory Configuration Mode (MCM[2:0] = 3)

After setting the MCM[2:0] to 3 the traffic store is organized in the following way:

Picture "copied" from [MVBC]

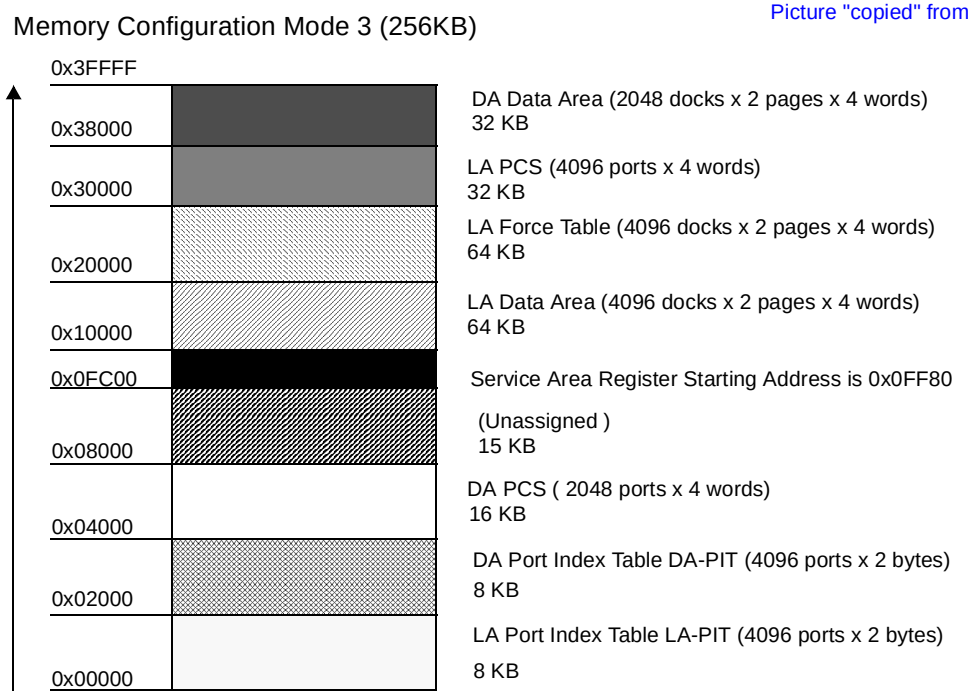


Figure 6: MVBC Start-up with MCM[2:0] = 3 for Class 2 or higher Devices [MVBC]

5.1.3. MVBC Register Summary - Service Area

Depending on the selected memory configuration mode, the MVBC internal registers - Part of the Service Area - do start at the traffic store address location:

- MCM[2:0] = 0. Default after Reset: 0x03F80
- MCM[2:0] = 1: 0x07F80
- MCM[2:0] = 2, 3 and 4: 0x0FF80

The Memory Control Register MCR is located at the address 0x0yF84 [MVBC]. The effective address in the traffic store depends on the MCM mode being set. Depending on the MCM[2:0] the value for "y" is "3", "7" or "F" as shown above [MVBC].

Address	Register
0yFF0h	**TR1**
0yFE0h	**TCR**
0yFD0h	**ECA**
0yFC0h	**ISR0**
0yFB0h	**IPR0**
0yFA0h	**MR**
0yF90h	**FC**
0yF80h	**SCR**

Address	Register
0yFF4h	**TR2**
0yFE4h	*(vacant)*
0yFD4h	**ECB**
0yFC4h	**ISR1**
0yFB4h	**IPR1**
0yFA4h	**MR2**
0yF94h	**EC**
0yF84h	**MCR**

Address	Register
0yFF8h	**TC1**
0yFE8h	*(vacant)*
0yFD8h	**DAOR**
0yFC8h	**IVR0**
0yFB8h	**IMR0**
0yFA8h	**DPR**
0yF98h	**MFR**
0yF88h	**DR**

Address	Register
0yFFCh	**TC2**
0yFECh	*(vacant)*
0yFDCh	**DAOK**
0yFCCh	**IVR1**
0yFBCh	**IMR1**
0yFACh	**DPR2**
0yF9Ch	**MFRE**
0yF8Ch	**STSR**

Figure 7: MVBC Register Summary - An Overview [MVBC]

5.2. KP99 DPM Interface Memory Map

5.2.1. KT94 Effective Address Calculations

Since the INTEL CPU used on the **KT94** segment addressing is being used. According the information found in [REF_02] the DPM interface uses the address segment 0xC800. The following effective address shall be assumed for the DPM interface (3 Examples are shown):

- Segment Start Address is 0xC800; DPM Address is 0x0000;
  Effective Address for the Intel CPU is: A[19:0] = 0xC8000
- Segment Start Address is 0xC800; DPM Address is 0x3000;
  Effective Address for the Intel CPU is: A[19:0] = 0xCB000
- Segment Start Address is 0xC800; DPM Address is 0x3FFF;
  Effective Address for the Intel CPU is: A[19:0] = 0xCBFFF

The different address locations inside the **KP99** interface memory area shall be understood as an offset. The effective address for the cpu inside the **KT94** must be calculated as shown above.

5.2.2. DPM To Traffic Store Access Mechanism - Windowing

The FPGA shall contain kind of an address translation mechanism which makes it possible to access all the different traffic memory location through small windows pointing into specific area of the traffic memory. Those configurable address windows shall operate in two different modes:

- Direct Mode. The start address for the corresponding window must be set through software - write op.
- Effective Mode. Reading the appropriate PIT location will update all needed address pointers. Therefore direct access to the different area in the traffic memory will be possible. In this mode the address calculations shall be done in hardware freeing the SW from this task.

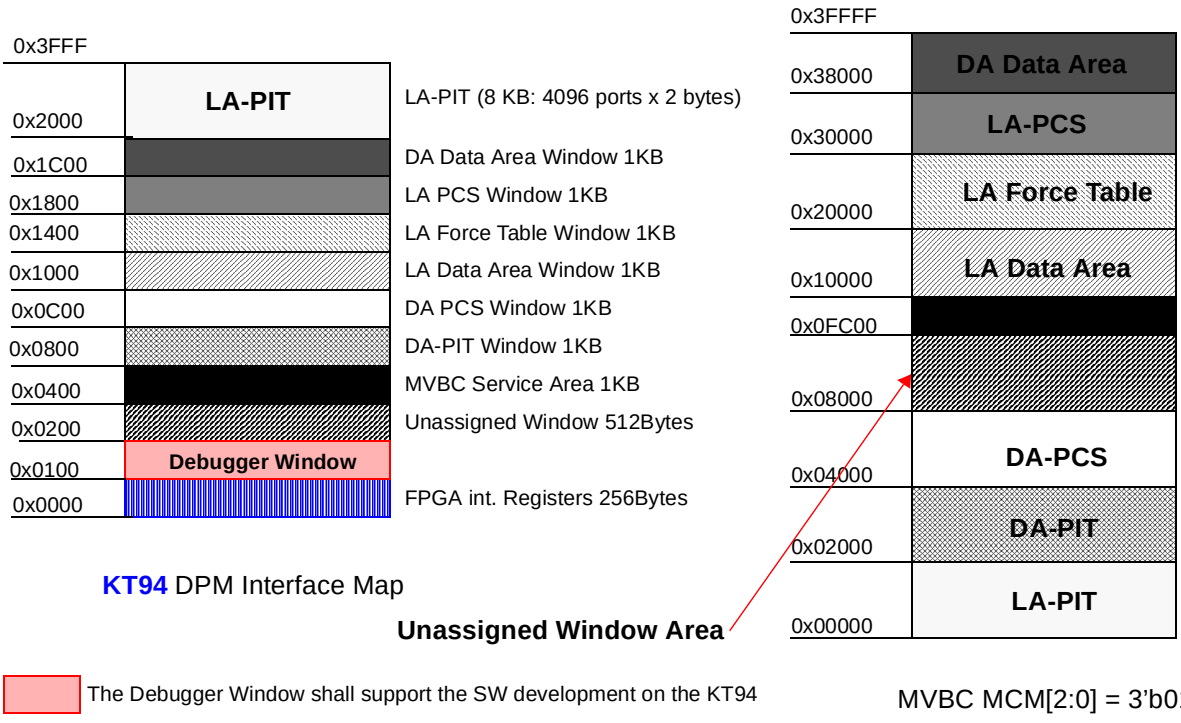


Figure 8: Traffic Memory Address Window Decomposition

The window mechanism shall be valid for the memory configuration mode 3. The MVBC service area must be accessible through all memory configuration modes.

5.2.3. DPM To Traffic Store Address Calculation

5.2.3.1. Port Index Tables

The traffic memory (TM) provides two Port Index Tables (PIT), one for Logical_Addressed_Data (used for process data) and one for Device_Addressed_Data (used to receive Device_Status_Reports). The contents of the PIT's is used as a further reference for the Port_Control_and_Status registers PCS. The configuration of the PCS determines if the port is a source or a sink port. Port Index PI=000 hex is used to assign all unused Logical and Device Addresses. If the memory configuration mode of the MVBC is set to MCM={2,3,4}, then every 16-bit word contains one 12-bit Port Index, covering bits 11..0. Bits 15..12 must be zero.

5.2.3.2. MVBC Address Evaluation from Port Index

The Port Index Table contents is used to compute the effective traffic memory addresses for the following TM memory blocks:

- PCS
- LA and DA Data Area
- Force Table (For logical address space only)

5.2.3.3. MVBC Address Evaluation Example:

Consider an incoming MF containing 0x0234 (F-Code=0, Port Address = 0x234). The MCM is set to 3 (256 KB Traffic Memory). First, the TM address 0x0468 is evaluated from the Port Address to read the Port Index Table (PIT) contents. Assuming, the addressed port index location does contain 0x000F, then the resulting addresses are computed as follows [MVBC]:

- PCS Address = PCS_Start_Addr(MCM = 3) + (0x000F shift_left 3) = 0x30000 + 0x0078 = 0x30078

Depending on the contents of the valid page flag VP inside the PCS register PCS_1[MVBC], the following offsets are computed:

- Offset(VP=0) = 0x00D8
- Offset(VP=1) = 0x00F8

Offset computation is done the following way (the example shown is: PIT-Contents is eq. to 0x000F):

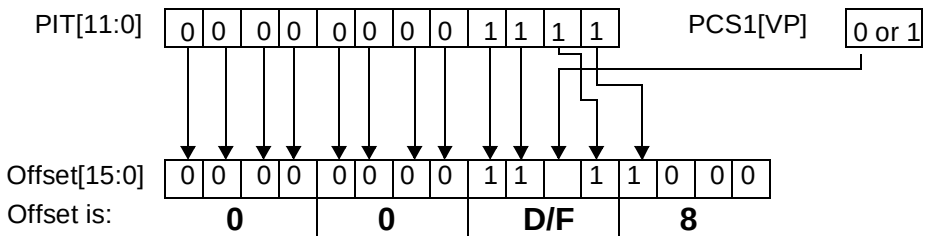


Figure 9: LA_DATA_Area and LA_Force_Table Offset Computation

- LA_Data_Area(VP=0) Address = 0x10000 + 0x00D8 = 0x100D8
- LA_Data_Area(VP=1) Address = 0x10000 + 0x00F8 = 0x100F8
- LA_Force_Table_Data(VP=0) Address = 0x200D8
- LA_Force_Table_Data(VP=1) Address = 0x200F8

Since not all the information is available by simply reading the PIT contents the address calculation unit on the **KP99** shall assume the VP-Bit in the PCS_1 Register being '0'. However the SW **has** to read the value of this flag in order know if an offset is needed or not.

5.2.3.4. KP99 Address Evaluation from Port Index

In order to access the different areas inside the traffic memory, some assumptions must be made. First the memory configuration mode of the MVBC chip must correspond with the memory configuration mode of the BAR_CSR (Base Address Control and Status Register). The contents of those three bits does control all the basic settings of the remaining base address registers. Therefore correct setting is mandatory! Second, the valid page flag within the BAR_CSR must be set according to the reading of the PCS_1[VP] contents. This will correct for the offset according to the offset computation shown above.q

5.3. FPGA Internal Registers

The registers inside the FPGA are 16-bit wide. However due to the 8 bit data interface two accesses must be made to read or write the registers. Within this document the registers will be presented in the 16-bit wide format.

5.3.1. Register Summary Part 1

These registers shall be implemented for compatibility reasons. Reading the locations shown below will allow the SW to recognize different hardware modules being connected on the interface of the KT94. The register mapping is done here for the **KP99**, however the order and the meaning of those registers have been made "compatible" with the ones defined in [REF_02].

Table 3: KP99 Register Overview Part 1

Name	Address	Short Description
AF_KP_GK	0x0000	• Devive Identification (Gerätekennung = "KP")
AF_KP_KN	0x0002	• Device Coupler Number (Kopplernummer "99")
AF_KP_RV	0x0004	• KP99 HW Version Number. Initial
AF_KP_LK	0x0006	• Hardbeat. The number shall increment all 20ms.(Lebenskennung Koppler)
Reserved Register Area 1	0x0008 - 0x001F	• Reserved, but **not** used for the KP99.
Reserved Register Area 2	0x0020 - 0x007D	• Reserved.
BAR_DBG	0x07E	• Base Address Register for the debug window.

5.3.2. The Base Address Register 'DEBUG' BAR_DBG

The Base Address Register 'DEBUG' BAR_DBG — Address(byte count): 0x07E

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BAR17	BAR16	BAR15	BAR14	BAR13	BAR12	BAR11	BAR10	BAR9	BAR8	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

• **BAR_DBG[17:8]**
The base address register 'Debug' shall map a specified memory area of the whole 256KB of traffic

memory into the available 256 bytes window available on the **KT94** DPM Interface map (page 17). This register does not depend on the LA-PIT reading! It does not depend on the MCM bits setting. This window shall support a debugger watching the whole traffic memory through the available 256 bytes window.

5.3.3. Register Summary Part 2

Table 4: KP99 Register Overview Part 2

Name	Address	Short Description
REG_CSR	0x080	• Main Control and Status Register
INT0_CSR	0x082	• MVBC INT0 Control and Status Register
INT1_CSR	0x084	• MVBC INT1 Control and Status Register
REG_DIAG	0x086	• Diagnostic Register
BAR_CSR	0x088	• Base Address Control and Status Register
BAR_0	0x08A	• Base Address Register 0. It points to the Unassigned Memory Window.
BAR_1	0x08C	• Base Address Register 1. It points to the MVBC Service Area.
BAR_2	0x08E	• Base Address Register 2. It points to the Device Address Port Index Table DA-PIT.
BAR_3	0x090	• Base Address Register 3. It points to the Device Address PCS Area.
BAR_4	0x092	• Base Address Register 4. It points to the LA Data Area.
BAR_5	0x094	• Base Address Register 5. It points to the LA Force Table.
BAR_6	0x096	• Base Address Register 6. It points to the Logical Address PCS Area.
BAR_7	0x098	• Base Address Register 7. It points to the DA Data Area.
PIT_DAT	0x0FE	• Contains the Value of the last LA-PIT Read Operation. The contents of this register does influence the other base address registers.

5.3.4. The Main Control and Status Register REG_CSR

Control and Status Register (REG_CSR): Address(byte count): 0x080

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0	ERS	EOP	-	-	-	-	RRG	RST
0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1
R	R	R	R	R	R	R	R	RW	RW	R	R	R	R	RW	RW

• **RST**

Whenever this flag is set the MVBC shall be reseted. It shall remain reseted until this flag is set back to zero.

• **RRG**

Whenever this flag is set the REGA's shall be reseted. It shall remain reseted until this flag is set back to zero.

• **EOP**

Enable Optical Port. Through this flag the communication through the optical interface can be enabled/ disabled. If it is disabled no information can come through the optical interface. It is decoupled from the rest of the circuitry.

• **ERS**

Enable RS485 Port. Through this flag the communication through the RS485 interface can be enabled/ disabled. If it is disabled no information can come through the RS485 connection. It is decoupled from the rest of the circuitry.

• **ID[7:0]**

FPGA hardware revision index. Starting with revision 1. Read only.

5.3.5. The MVBC INT0 Control and Status Register INT0_CSR

MVBC INT0 Control and Status Register INT0_CSR Address(byte count): 0x082

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	TST0	INT0	ENI0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	RW	R	RW

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

• **ENI0**

Enable Interrupt 0. Whenever this flag is set - set to '1' - the interrupt request line INT0 from the MVBC (pin 27) shall be routed to the /INT signal on J1 (page 10). Whenever this flag is reseted - set to '0' - the interrupt request line from the MVBC shall be masked towards the signal /INT on J1. In such a case a logic '0' on INT0 from the MVBC shall not drive the /INT signal low. The signal /INT shall remain logic '1'.

• **INT0**

Interrupt 0 status and control flag. Reading this flag shall reflect the actual level on the MVBC signal/ pin INT0. This can be used in SW to use the MVBC internal interrupt controller in polling mode [MVBC].

• **TST0**

Whenever TST0 is a logic '1' the interrupt request line /INT on the connector J1 shall be asserted - set to '0'. This shall be used to test and verify the interconnection without the usage of the MVBC. The line

/INT on J1 shall remain asserted until the state of the flag TST0 has been reseted by an additional write operation with a logic '0'.

5.3.6. The MVBC INT1 Control and Status Register INT1_CSR

MVBC INT1 Control and Status Register INT1_CSR — Address(byte count): 0x084

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	TST1	INT1	ENI1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	RW	R	RW

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

- **ENI1**

Enable Interrupt 1. Whenever this flag is set - set to '1' - the interrupt request line INT1 from the MVBC (pin 28) shall be routed to the /INT signal on J1 (page 10). Whenever this flag is reseted - set to '0' - the interrupt request line from the MVBC shall be masked towards the signal /INT on J1. In such a case a logic '0' on INT1 from the MVBC shall not drive the /INT signal low. The signal /INT shall remain logic '1'.

- **INT1**

Interrupt 1 status and control flag. Reading this flag shall reflect the actual level on the MVBC signal/ pin INT1. This can be used in SW to use the MVBC internal interrupt controller in polling mode [MVBC].

- **TST1**

Whenever TST1 is a logic '1' the interrupt request line /INT on the connector J1 shall be asserted - set to '0'. This shall be used to test and verify the interconnection without the usage of the MVBC. The line /INT on J1 shall remain asserted until the state of the flag TST1 has been reseted by an additional write operation with a logic '0'.

5.3.7. The Main Diagnostic Register REG_DIAG

Main Diagnostic Register REG_DIAC — Address(byte count): 0x086

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	SYNC	SF	OC	ICB	ICA	-	-	ERR
0	0	0	0	0	0	0	0	0	?	?	?	?	0	0	0
R	R	R	R	R	R	R	R	RW0	R	R	R	R	RW0	RW0	RW1

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

- **ERR**

Error LED. When this flag is set then the error LED driver output shall be turned on - set to '0'.

• ICA, ICB

MVBC manchester code inputs. The voltage level on those lines can be read back. For diagnostic purpose only.

• OC

MVBC manchester code output. The voltage level on this output can be read back. For diagnostic purpose only.

• SF

MVBC send frame output. The voltage level on this output can be read back. For diagnostic purpose only.

• SYNC

Strobe - pin 2 - output signal from the MVBC. This flag represents the inverted level from the MVBC output STROBE_N. However there is an additional edge detection and latching logic integrated, which will latch the occurrence of a strobe pulse on the MVBC [MVBC]. Writing this flag with '0' will reset the 'hidden' latch logic. For diagnostic purpose only.

5.3.8. The Base Address Control and Status Register BAR_CSR

The Base Address Control and Status Register BAR_CSR Address(byte count): 0x088

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	VP	x	x	x	x	x	MCM2	MCM1	MCM0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	RW	-	-	-	-	-	RW	RW	RW

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

• MCM[2:0]
Memory configuration mode. Those bits shall behave according to the description in [MVBC]. Those bits must correspond with the settings inside the MCR Register [MVBC]! The value of those bits preset all the other base address registers according to the traffic memory decomposition found in the description of the MVBC ASIC [MVBC].

• VP
The valid page flag. This flag shall influence the offset calculation for the data area. It shall have exactly the same meaning as the VP Flag inside the PCS_1 Register of the traffic store. Setting/Resetting this flag shall adjust the starting address of the data area as mentioned above (page 18).

5.3.9. The Base Address Register ’0’ BAR_0

The Base Address Register ’0’ BAR_0 — Address(byte count): 0x08A

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BAR14	BAR13	BAR12	BAR11	BAR10	BAR9	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return “0”.

- **BAR_0[15:0]**

The base address register ’0’ shall map the unassigned memory area of the traffic memory into the required 512 bytes window available on the **KT94** DPM Interface map (page 17). The starting address depends on the MCM settings within the BAR_CSR register [eq. to the MCR of the MVBC]. The mapping is done in 512 bytes increments. The effective address being computed is the start address - depends on the MCM[2:0] - plus the number BAR[14:9] of 512 bytes increments. This register does not depend on the LA-PIT reading!

5.3.10. The Base Address Register ’1’ BAR_1

The Base Address Register ’1’ BAR_1 — Address(byte count): 0x08C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return “0”.

- **BAR_1[15:0]**

This register has been skipped, since the service area of the MVBC maps well into the 1KB window available. The starting address depends on the MCM[2:0] settings only!

5.3.11. The Base Address Register ’2’ BAR_2

The Base Address Register ’2’ BAR_2 — Address(byte count): 0x08E

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	BAR12	BAR11	BAR10	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	RW	RW	RW	R	R	R	R	R	R	R	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return “0”.

• **BAR_2[15:0]**
The base address register ’2’ shall map the DA-PIT table area of the traffic memory into the required 1KB window available on the **KT94** DPM Interface map (page 17). The starting address depends on the MCM settings within the BAR_CSR register [eq. to the MCR of the MVBC]. The mapping is done in 1KB increments. The effective address being computed is the start address - depends on the MCM[2:0] - plus the number BAR[12:10] of 1KB increments. This register does not depend on the LA-PIT reading!

5.3.12. The Base Address Register ’3’ BAR_3

The Base Address Register ’3’ BAR_3 — Address(byte count): 0x090

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	BAR13	BAR12	BAR11	BAR10	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return “0”.

• **BAR_3[15:0]**
The base address register ’3’ shall map the DA-PCS area of the traffic memory into the required 1KB window available on the **KT94** DPM Interface map (page 17). The starting address depends on the MCM settings within the BAR_CSR register [eq. to the MCR of the MVBC]. The mapping is done in 1KB increments. The effective address being computed is the start address - depends on the MCM[2:0] - plus the number BAR[13:10] of 1KB increments. This register shall not depend on the LA-PIT reading nor shall it depend on the DA-PIT reading!

5.3.13. The Base Address Register ’4’ BAR_4

The Base Address Register ’4’ BAR_4 — Address(byte count): 0x092

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BAR15	BAR14	BAR13	BAR12	BAR11	BAR10	BAR9	BAR8	BAR7	BAR6	BAR5 + VP	BAR4	BAR3	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return “0”.

• **BAR_4[15:0]**
The base address register ’4’ shall map the LA-DATA area of the traffic memory into the required 1KB window available on the **KT94** DPM Interface map (page 17). The **starting address** shall depend on the MCM settings within the BAR_CSR register [eq. to the MCR of the MVBC]. The effective address being computed shall be the **start address**. It shall depends on the LA-PIT reading - PIT_DAT register contents - the MCM[2:0] value and the value of the valid page bit - BAR_CSR[VP] (see on page 18). The result of this computation shall be stored into BAR_4 before the access is being made. It shall contain the effective address within the choosen 64KB window [MVBC].

NOTE: The user shall under normal circumstances **not** change the contents of this register. The contents of the register BAR_4[15:3] being computed **always** points to the required starting point. However in some cases it may be useful to modify the start address directly i.e direct mode.

5.3.14. The Base Address Register '5' BAR_5

The Base Address Register '5' BAR_5 — Address(byte count): 0x094

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BAR15	BAR14	BAR13	BAR12	BAR11	BAR10	BAR9	BAR8	BAR7	BAR6	BAR5 + VP	BAR4	BAR3	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

• **BAR_5[15:0]**
The base address register '5' shall map the LA-FORCE-TABLE area of the traffic memory [MVBC] into the required 1KB window available on the **KT94** DPM Interface map (see page 17). The **starting address** shall depend on the MCM settings within the BAR_CSR register [eq. to the MCR of the MVBC]. The effective address being computed shall be the **start address**. It shall depends on the LA-PIT reading - PIT_DAT register contents - the MCM[2:0] value and the value of the valid page bit - BAR_CSR[VP] (see on page 18). The result of this computation shall be stored into BAR_5 before the access is being made. It shall contain the effective address within the chosen 64KB window [MVBC].

NOTE: The user shall under normal circumstances **not** change the contents of this register. The contents of the register BAR_5[15:3] being computed **always** points to the required starting point. However in some cases it may be useful to modify the start address directly i.e direct mode.

5.3.15. The Base Address Register '6' BAR_6

The Base Address Register '6' BAR_6 — Address(byte count): 0x096

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BAR14	BAR13	BAR12	BAR11	BAR10	BAR9	BAR8	BAR7	BAR6	BAR5	BAR4	BAR3	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

• **BAR_6[15:0]**
The base address register '6' shall map the LA-PCS area of the traffic memory [MVBC]into the required 1KB window available on the **KT94** DPM Interface map (see page 17). The **starting address** shall depend on the MCM settings within the BAR_CSR register [eq. to the MCR of the MVBC]. The effective address being computed shall be the **start address**. It shall depends on the LA-PIT reading - PIT_DAT register contents - and on the MCM[2:0] value (see on page 18). The result of this computation shall be stored into BAR_6 before the access is being made. It shall contain the effective address within the chosen 64KB window [MVBC].

NOTE: The user shall under normal circumstances **not** change the contents of this register. The contents of the register BAR_6[15:3] being computed **always** points to the required starting point. However in some cases it may be useful to modify the start address directly i.e direct mode.

5.3.16. The Base Address Register '7' BAR_7

The Base Address Register '7' BAR_7 — Address(byte count): 0x098

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BAR14	BAR13	BAR12	BAR11	BAR10	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

• **BAR_7[15:0]**
The base address register '7' shall map the DA-DATA area of the traffic memory into the required 1KB window available on the **KT94** DPM Interface map (page 17). The starting address depends on the MCM settings within the BAR_CSR register [eq. to the MCR of the MVBC]. The mapping is done in 1KB increments. The effective address being computed is the start address - depends on the MCM[2:0] - plus the number BAR[14:10] of 1KB increments. This register shall not depend on the LA-PIT reading nor shall it depend on the DA-PIT reading nor shall it depend on the VP-Bit of the DA-PCS1 register!

5.3.17. The LA Port Index Table Data Register PIT_DAT

The LA Port Index Table Data Register PIT_DAT — Address(byte count): 0x0FE

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	PIT11	PIT10	PIT9	PIT8	PIT7	PIT6	PIT5	PIT4	PIT3	PIT2	PIT1	PIT0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Note: Writing to unimplemented bits has no effect. Reading them will always return "0".

• **PIT[15:0]**
The register shall contain the value read back from the last LA-PIT table reading. The contents of this register will adjust the other registers. However for diagnostic and testing purpose, it shall be possible to directly write into this register.

6. Mechanics

The **KP99** interface shall be mounted directly next to the KT94 SPS. Fixation shall be made through a mounting possibility on a standard DIN "Schiene" at the back side. The case itself shall be metal case for better EMI compatibility. However no electronic voltage shall be connected to the case directly except the RS485 cable shield (page 11). The thickness of the metal case shall be ~300um.

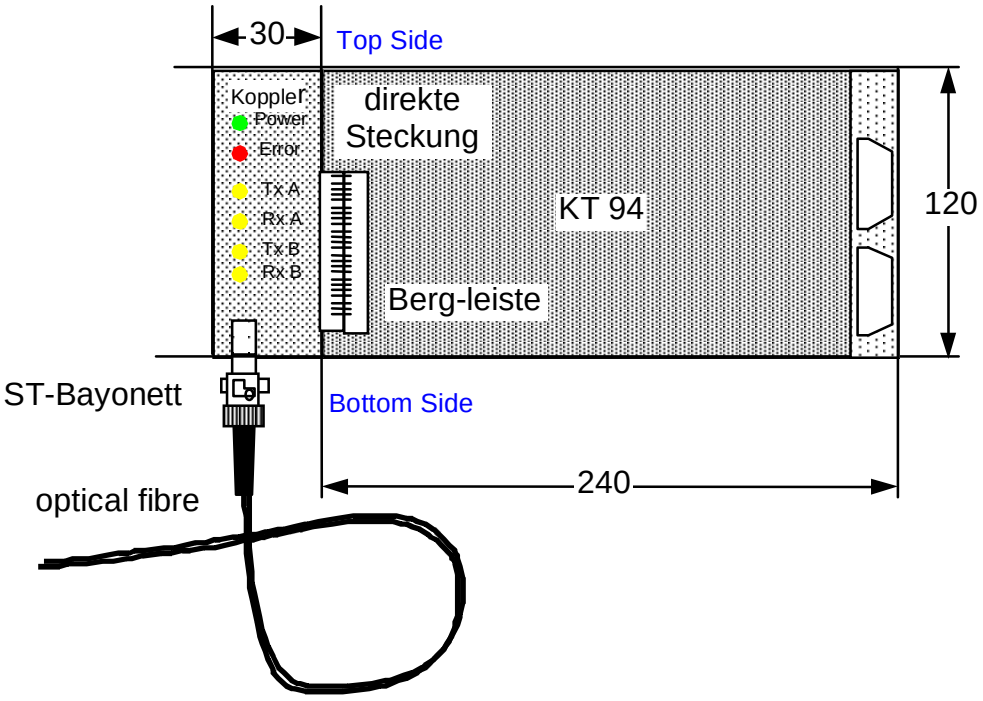


Figure 10: **KP99** and KT94 mounting and connection

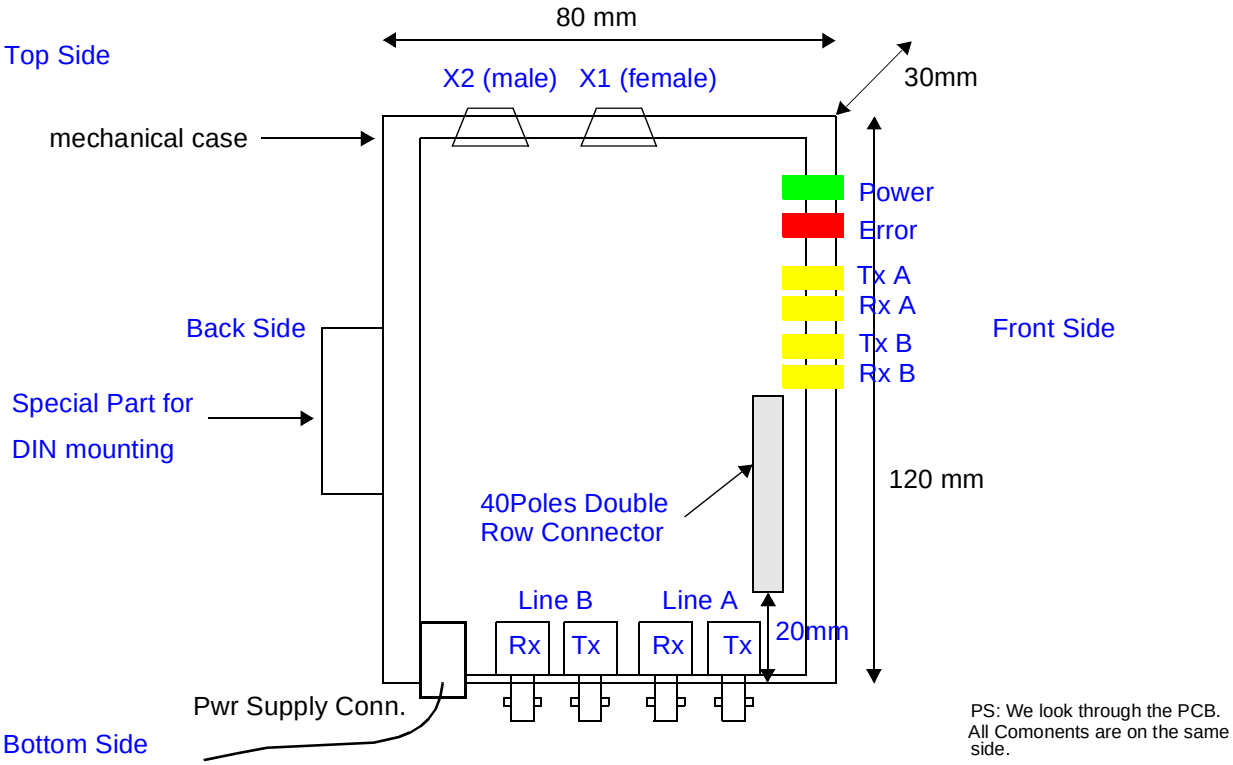


Figure 11: Some mechanical issues concerning placement and interfacing

7. Operational Conditions and Design Constraints

7.1. Operating Conditions

7.1.1. Supply Voltage

+24Volt DC +- 25% deviation.

7.1.2. Supply Ripple

Not more than 500mVolt RMS

7.1.3. Supply Current Requirements for the **KP99**

Average power consumption ~2000 mW.

7.1.4. Temperature Range

Operation beyond following limits may impair the lifetime of the module.

- Operating Temperature: -10...+55°Celsius
- Storage Temperature: -40...+85°Celsius

7.2. Startup Power-OFF and Reset Behavior

- When the internal +5V supply has been stabilized, an additional 500msec is allowed before the FPGA device must be operational!
- Reseting the interface while the supply voltage is not switched off and on again, shall not require any delay after reset negation before the FPGA is operational again.
- Power interruptions more than 50msec may not guarantee proper operation anymore. A local voltage monitor shall assert the reset line, when the supply voltage falls below the tolerated minimum supply level.

7.3. Handling

The module is to be considered as an ESD sensitive part. ESD handling requirements have to be observed. Life insertion of the module is not supported!

7.4. Testing

7.4.1. ICT and JTAG

The signal names starting with JTAG shall be for production test only. The following devices shall be accessible through JTAG:

- MVBC
- FPGA

All the other nodes shall be made available for standard in-circuit testing.

7.4.2. Functional

In order to fully test the interface module **KP99** functional testing shall be foreseen as well. A source file library written in C can be provided. However it will need some adaption due to hardware and compiler dependencies.

7.5. FPGA Design Methodology

The functional description for the FPGA shall be done using the Verilog Hardware Description Language. Normal design flow will go through design entry - text based -, simulation - behavioral -, synthesis, place and route and again gate level simulation for design verification.

7.6. Component Selection

The following components shall be used within the design:

- MVBC02A or higher.
- The REGA ASIC from ADtranz.
- Opto couplers from HP. Serie HCPL-06xx. This is because of speed constraints.
- The fibre optical components from HP. Serie HFBR-x4x4. Those components have been gone through a selection process.

8. Revision history

This is the initial version of this document.

9. Document Approval

The document has been reviewed and approved by ABB Industrie AG in Dättwil.

Date, Place Signature