Specification for HTPA8x8L5.5M(LC)

Rev4: 2013.02.22 Fg

The HTPA8x8L_/_M(LC) is a fully calibrated, low cost thermopile array module, with fully digital SPI interface. The module delivers an electrical offset and ambient temperature compensated output stream, which can be already used for image processing, pattern recognition and presence detection purposes. Object temperatures can be easily obtained by this data stream, a look up table and the calibrated sensitivity constants, which can be found in the EEPROM of the module.

Order Code Example

HTPA32x31L10/1.0HiM(SPI)

Interface: SPI→ SPI device (14bit ADC)
LC→SPI, 12bit ADC, low speed, external processing required
UDP→Ethernet, CAT5 cable connection
UART→RS232-like, Level: 3.3V

Type: A→Application set: comes with GUI, housing, power supply
M→Module: HTPA sensor soldered to PCB, calibrated stream
S→Sensor: HTPA sensor only. Analogous output.

Sensitivity: Hi→Increased sensitivity
Without "Hi"→ Standard sensitivity

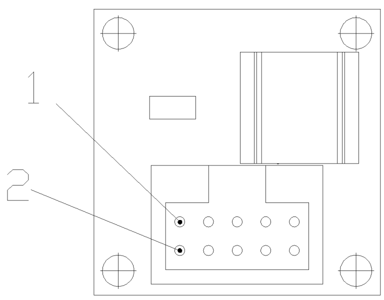
Optics: L→focal length: In example L10 = 10 mm focal length.
/→ F-Number: In example /0.8
For optics see also "HTPA standard optics"

Type: HTPA32x31 (Please contact support for all available HTPA and module combinations.

For modules, the recommended type is M(SPI). The advantages are the better ADC resolution, wider input voltage range, wider measurement range.

Pinout

Pin Assignment HTPA32x31M(LC)			
Pin	Name	Description	Type
1	#MCLR	Master clear, negotiated	Digital Input
2	VDD	Positive supply voltage	Power
3	VSS	Negative supply voltage	Power
4	VSS	Negative supply voltage	Power
5	#SS	Slave select, negotiated	Digital Input
6	SDO	Serial data out of module	Digital Output
7	SDI	Serial data in of module	Digital Input
8	SCK	Serial clock	Digital Input
9	MCLK	Master clock, drives HTPA sensor	Digital Input
10	#VD	Valid Data, negotiated.	Digital Output



HEIMANN Sensor GmbH
Grenzstr. 22
D-01109 Dresden / Germany

Contact / Customer Support
Phone 49 (0) 6123 60 50 30
Fax 49 (0) 6123 60 50 39

Internet
www.heimannsensor.com
mail: info@heimannsensor.com

SPI Interface:

SCK-Frequency:	350 kHz …10 MHz

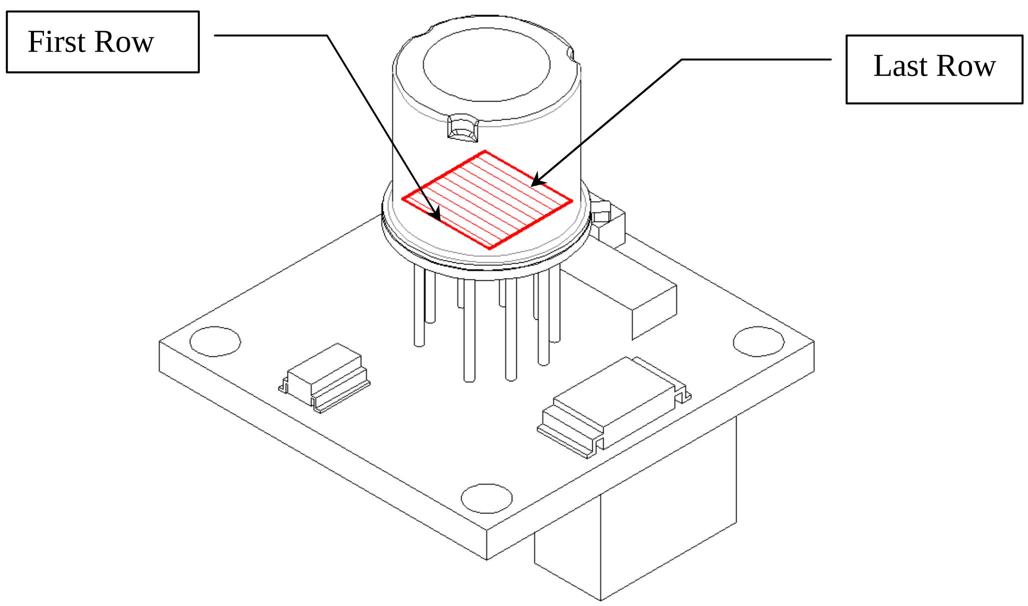
Protocol Specifications:

Data format:	16 data bits
Frame Sync:	None
Module-Selection:	$\overline{\text{SS}}$-Pin
Clock Edge Select:	Serial output data changes on transition from idle to active clock state
SPI Data Input Sample Phase:	Data sampled on transition from active to idle clock state
Clock Polarity:	Idle State is high level, active is low level.

Electrical Specifications:

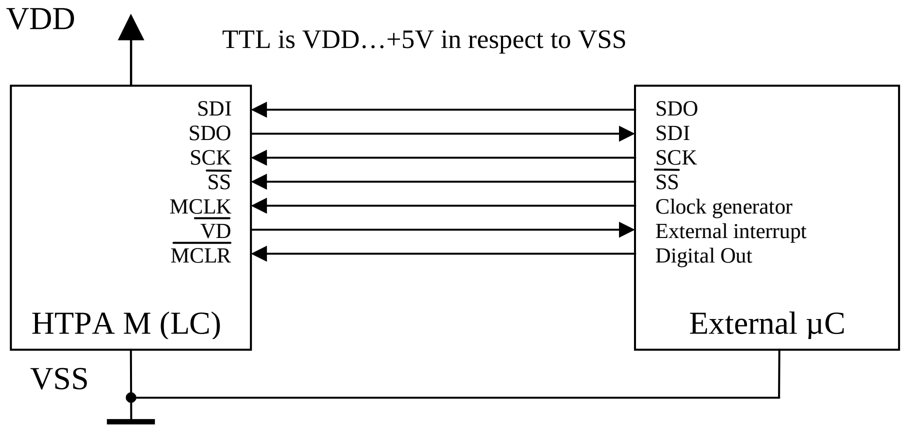
VDD:	Supply (+5.0V DC)
SPI Transmit/Receive:	TTL
VSS	GND
Power Supply:	**5.0** VDC +/- 2%, 300mA
IDD (Idle mode)	20 mA
IDD (Operating mode)	45 mA

HTPA8x8L5.5M(LC) Optical Orientation of Pixels:

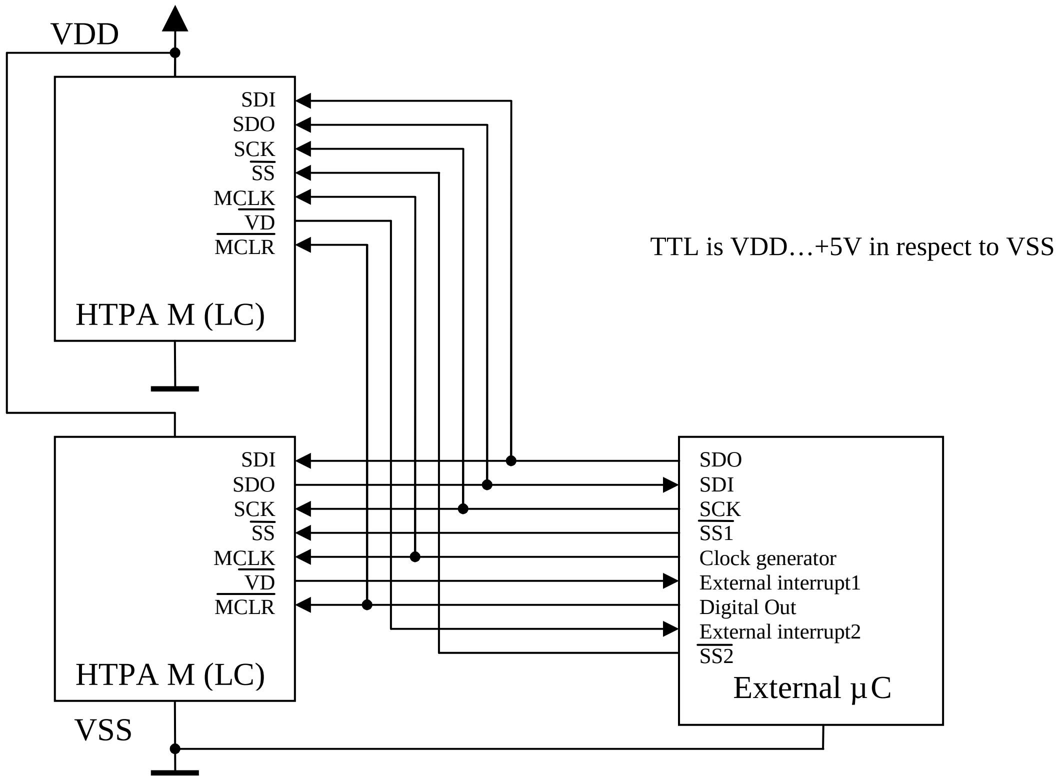


Electrical Connections:

Single Module:

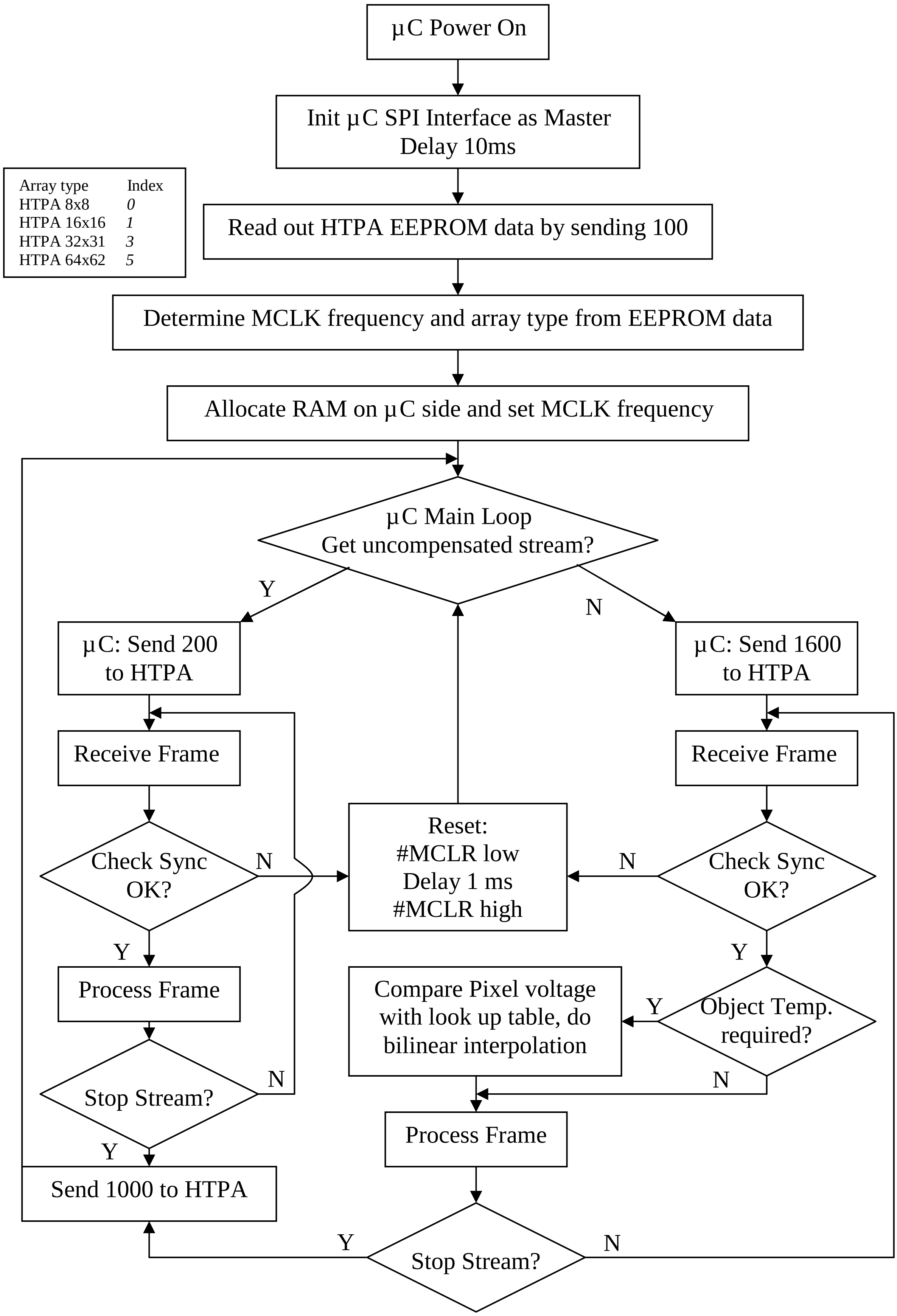


Multiple Modules (preliminary):



Communication and Timings:

Proposed flow chart of communication. (Master is referred as µC, Slave as HTPA module)



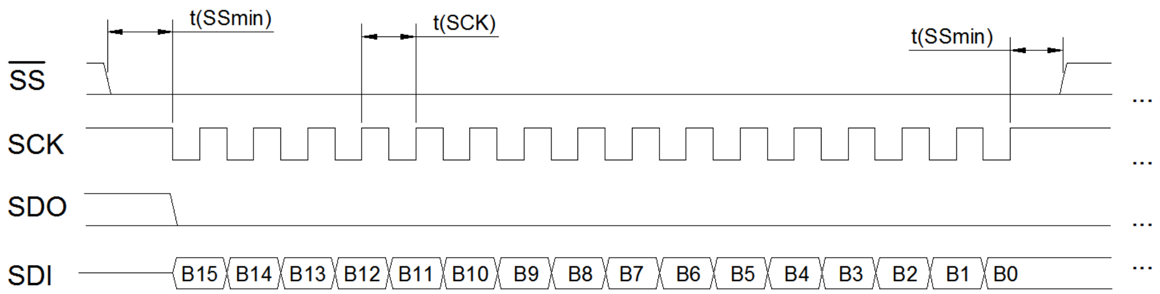
HEIMANN Sensor GmbH
Grenzstr. 22
D-01109 Dresden / Germany

Contact / Customer Support
Phone 49 (0) 6123 60 50 30
Fax 49 (0) 6123 60 50 39

Internet
www.heimannsensor.com
mail: info@heimannsensor.com

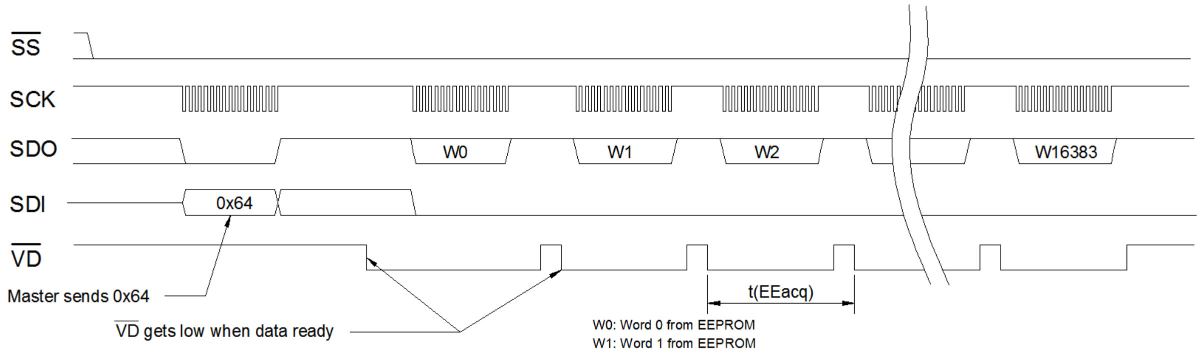
Communication and Timings (continuation):

Receive of command:

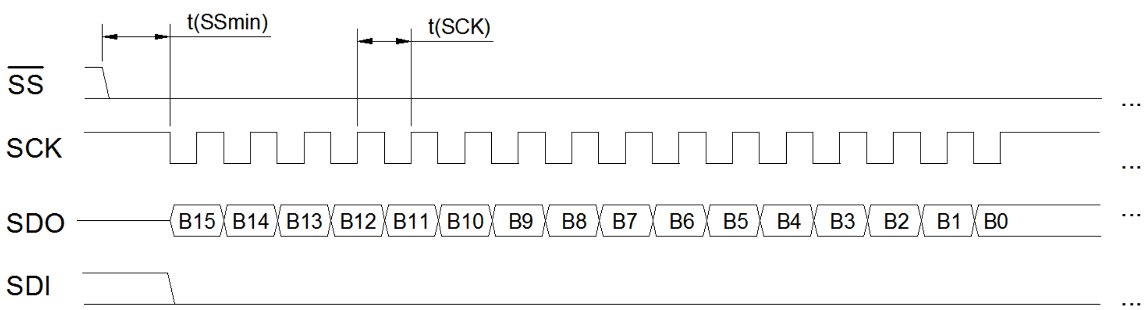


(High state of #SS is not necessary, only for communication with multiple devices)

Send of EEPROM content:

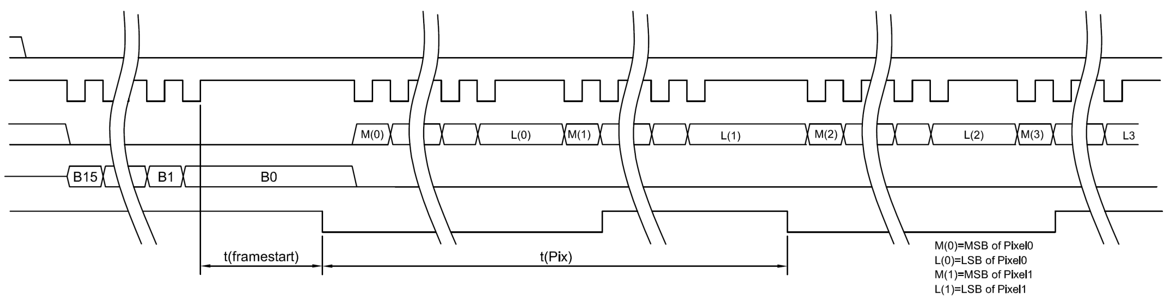


Pixel data:



B15...B0: Raw or compensated ADC reading (depending from streaming mode)

Receive of stream command:



For streaming the adequate frequency needs to be applied to the MCLK pin of the module.

Communication and Timings (continuation):

Absolute values:

	MIN	NOM	MAX	Unit	Remarks
MCLR pulse width (low)	2			µs	
t(SSmin)	150			ns	
t(SCK)	0.1	1	2.86	µs	
t(EEacq)	185			µs	
t(framestart)		29		ms	f(MCLK)=1 MHz
t(Pix)		208		µs	f(MCLK)=1 MHz

t(Pix) and t(framestart) depend on the given MCLK frequency of the master. In example: MCLK frequency is 1003 kHz, then t(Pix) and t(framestart) is calculated via

$$t(Pix) = \frac{208}{f(MCLK)} = \frac{208}{1003000} = 207{,}4\mu s \qquad t(framestart) = t(Pix) \cdot 9 \cdot 8 + 14\text{ms} = 29\text{ms}$$

Important:
The SCK frequency needs to be at least that large, that the 16 bits can be submitted within tPix. Therefore, the following condition must be always true:

$$\boxed{16 \cdot t(SCK) < t(Pix)}$$

EEPROM Mapping:

Overview:

Start address	End address	Data type	Value
0x0	0x3	float	Minimum value of PixC's for scaling
0x4	0x7	float	Maximum value of PixC's for scaling
0x8	0x9		Heimann Sensor reserved
0xA	0xA	char	Table number
0xB	0x33		Heimann Sensor reserved
0x34	0x37	float	PTATgrad
0x38	0x3B	float	PTAToff
0x3C	0x58		Heimann Sensor reserved
0x59	0x5A	unsigned int	MCLK Frequency in kHz
0x5B	0x79		Heimann Sensor reserved
0x80	0xFF	unsigned int	scaled down values of PixC's
0x100	0x3FFF		Heimann Sensor reserved

Important Note:

unsigned int: 2 byte; float: 4 byte; char: 1 byte
All the values are stored (if larger than one byte) in little endian, the so called „Intel-Format“.
Example for the MCLK-Frequency:

$$MCLK_{LB} = \text{EEPROM}[0x59] \quad MCLK_{HB} = \text{EEPROM}[0x5A]$$
$$MCLK = 256 \cdot MCLK_{HB} + MCLK_{LB}$$

EEPROM Mapping (continuation):

Details for PixC's:

Start address	End address	Data type	Value
0x80	0x81	unsigned int	scaled PixC value of Pixel 0
0x82	0x83	unsigned int	scaled PixC value of Pixel 1
0x84	0x85	unsigned int	scaled PixC value of Pixel 2
…	…	…	…
0xFE	0xFF	unsigned int	scaled PixC value of Pixel 63

Calculation of the PixC's:

1. Determine minimum and maximum value of the PixC's out of the EEPROM data by reading associated EEPROM value into a float constant. Pseudocode in C, see function "getPixC(void);"
2. Now scale all scaled down PixC's out of the EEPROM content back to their original value and store them in RAM of your system.

Formulas:

$$PixC_{MAX} = \mathrm{EEPROM}[0x0 - 0x3] \quad \text{(4 byte float value in little endian)}$$
$$PixC_{MIN} = \mathrm{EEPROM}[0x4 - 0x7] \quad \text{(4 byte float value in little endian)}$$

$$PixC(PixelX) = \frac{\mathrm{EEPROM}[0x80 + (\mathrm{X} \cdot 2)] \cdot (PixC_{MAX} - PixC_{MIN})}{65535} + PixC_{MIN}$$

```
unsigned int PixC[992];     //The scaled back PixC's. Most likely, this should be global.

void getPixC(void)          //this function detemines the pixel constants. Precondition: EEPROM content is stored in the char array "EEPROM"
{
        float common[2],min,max;
        unsigned int addr=0x80,i;   //the start address for the scaled pixel constants
        unsigned int pcl;           //this stores the two bytes from the scaled down PixC out of EEPROM.

        memcpy((char*)&common,(unsigned char*)&EEPROM[0],sizeof(float)*2);        //the address of the scaling values for the pixc's
        min=common[0];
        max=common[1];
        for(i=0;i<PIXEL;i++){
        memcpy((char*)&pcl,(unsigned char*)&EEPROM[addr],2);        //include string.h for memcpy
        addr+=2;
        PixC[i]=(unsigned int)(((float)pcl/65535.0)*(max-min)+min+0.5);
        }

        return;
}
```

Serial order of data in stream:

Compensated Voltage Mode	
Dataset	Value
0	offset corrected Voltage of Pixel0 in in digits
1	offset corrected Voltage of Pixel1 in in digits
2	offset corrected Voltage of Pixel2 in in digits
…	…
63	offset corrected Voltage of Pixel63 in in digits
64	elOff0 in digits+0x7000
65	elOff1 in digits+(0x800<<4)
66	elOff2 in digits+(0x90<<8)
67	elOff3 in digits+(0xA<<12)
68	PTAT0 in digits | (TA&0xF000)
69	PTAT1 in digits | ((TA&0x0F00)<<4)
70	PTAT2 in digits | ((TA&0x00F0)<<8)
71	PTAT3 in digits | ((TA&0x000F)<<12)

Raw Voltage Mode	
Dataset	Value
0	absolute Voltage of Pixel0 in in digits
1	absolute Voltage of Pixel1 in in digits
2	absolute Voltage of Pixel2 in in digits
…	…
63	absolute Voltage of Pixel63 in in digits
64	elOff0 in digits+0x7000
65	elOff1 in digits+(0x800<<4)
66	elOff2 in digits+(0x90<<8)
67	elOff3 in digits+(0xA<<12)
68	PTAT0 in digits
69	PTAT1 in digits
70	PTAT2 in digits
71	PTAT3 in digits

Each dataset consists of a 16 bit value. The 16 bit values are transmitted with MSB first. In case of compensated voltage mode a signed 16 bit value is transmitted, in case of raw voltage mode an unsigned 16 bit value. Signed values are always in 2's complement. In both modes the module transmits a control word in the MSB's of the datasets 63…67. (0x789A), this can be used for syncing. In compensated mode, the ambient temperature TA (unsigned int, 16 bit, deci-Kelvin) is also transmitted in the MSB's of the datasets 68…72.

Pixel Map:

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

Communication commands:

Sent Command	Answer / Result
100	Output of EEPROM content. Data ready of each 2 bytes is signified by #VD pin.
200	Module streams out uncompensated, raw data stream. Data ready of each 4 bytes is signified by #VD pin.
1000	Stops streaming mode of module.
1600	Module streams offset corrected stream (electrical and thermal). Data ready of each 4 bytes is signified by #VD pin.

Precondition for all streaming modes:

MCLK signal is generated and frequency is in limits shown by the section “Absolute Maximum Ratings”

Preconditions for compensated streams
MCLK signal is generated and frequency matches EEPROM content. Failure of MCLK should be < ± 3%
VDD must be in the given limits (5V +/-2%). False values for these two may affect calculated absolute object temperatures. False values for the MCLK frequency also may result in pattern formation in frame.

Absolute Maximum Ratings:

Value	MIN	NOM	MAX	Unit	Remarks
TTL Frequency on pin MCLK	MCLK-3%	MCLK	MCLK+3%	Hz	in compensated streaming mode
TTL Frequency on pin MCLK	0.1		1.7	MHz	in raw voltage streaming mode
VDD in respect to VSS	-0.3	5	6.5	V	
VDD in streaming mode	4.9	5	5.1	V	False VDD values affect compensation
Voltage on digital pin with respect to VSS	-0.3		VDD+0.3	V	
Current consumption	37	45	50	mA	In streaming
Current consumption	18	20	25	mA	Idle

Temperature Calculation:

1. Init SPI Interface
2. Read out EEPROM data
3. Determine MCLK frequency, apply to MCLK pin (Refer to EEPROM Mapping)
4. Determine pixel constant PixC for each sensitive pixel, keep them in RAM (Refer also to EEPROM mapping)
5. Enable ISR connected to the #VD pin of the module
6. Write 1600 via the SPI interface to the module
7. Module starts to run and signifies valid data with pull down of #VD
8. In the ISR get 16 bit within the given timings from the module
9. This word represents the compensated pixel voltage of the corresponding pixel. For serial order of the pixels in frame refer to “Serial order of data in stream”
10. Scale the pixel sensitivity according to the following formula, using the PixC’s:

$$V_S(X) = \frac{1E8 \cdot V_C(X)}{PixC(X) \cdot \varepsilon}$$

    Where ε is the emissivity of the object, $V_S(X)$ is the sensitivity corrected voltage of pixel X, $V_C(X)$ is the offset compensated voltage of pixel X (submitted by the module).

11. Compare the $V_S(X)$ value with the pixel voltages in the look up table (vertical axis)
12. Calculate the ambient temperature of the sensor out of the given values from the module (see “Serial order of data in stream”). This pseudocode may be used for ambient temperature calculation:

```
unsigned int getTA(void)      //this function detemines the current ambient temperature TAmb.
{                             // Precondition: The fetched data was written to the global array RECEIVEBUFFER[72]
        unsigned int TAmb;

        TAmb=ReceiveBuffer[68]&0xF000;
        TAmb|=((ReceiveBuffer[69]&0xF000)>>4);
        TAmb|=((ReceiveBuffer[70]&0xF000)>>8);
        TAmb|=((ReceiveBuffer[71]&0xF000)>>12);

        return TAmb;
}
```

13. Compare the T_{AMB} value with the horizontal axis of the look up table.
14. Do a bilinear interpolation of the 4 neighbour supporting points, where T_{AMB} and $V_S(X)$ intersect.
15. The result is the object temperature in deci-Kelvin [dK].

C-Code for all these calculations can be found in our SDK (**S**oftware **D**evelopment **K**it). Furthermore, the SDK is able to fetch the data from the module and sends it to our GUI (**G**raphical **U**ser **I**nterface) which can visualize the data, records videos and text files and has many additional features. For more information see www.heimannsensor.com.

HEIMANN Sensor
HEIMANN SENSOR GMBH

Look up table:

Pixel voltag	Ambient Temperature [dK] 2582	2732	2882	3032	3182	3332	3482
-384	0x000	0x000	0x000	0x000	1643	2315	2698
-320	0x000	0x000	0x000	1483	2201	2585	2880
-256	0x000	0x000	1500	2143	2506	2789	3032
-192	0x000	1664	2148	2468	2727	2955	3164
-128	1876	2210	2471	2698	2904	3097	3281
-64	2311	2512	2701	2880	3053	3221	3386
0	2582	2732	2882	3032	3182	3332	3482
64	2786	2908	3034	3164	3297	3433	3571
128	2953	3056	3165	3281	3401	3525	3653
192	3095	3185	3282	3386	3496	3611	3730
256	3219	3300	3387	3482	3583	3690	3802
320	3331	3403	3483	3571	3665	3765	3871
384	3431	3498	3572	3653	3741	3835	3936
448	3524	3585	3654	3730	3813	3902	3997
512	3609	3667	3731	3802	3881	3965	4056
576	3689	3743	3803	3871	3945	4026	4113
640	3764	3814	3872	3936	4006	4083	4167
704	3834	3882	3936	3997	4065	4139	4219
768	3901	3946	3998	4056	4121	4192	4269
832	3964	4008	4057	4113	4175	4243	4318
896	4025	4066	4113	4167	4227	4292	4364
960	4083	4122	4168	4219	4276	4340	4410
1024	4138	4176	4220	4269	4325	4386	4454
1088	4191	4228	4270	4318	4371	4431	4496
1152	4242	4278	4318	4364	4416	4474	4538
1216	4292	4326	4365	4410	4460	4516	4578
1280	4339	4372	4410	4454	4502	4557	4617
1344	4385	4417	4454	4496	4544	4597	4655
1408	4430	4461	4497	4538	4584	4636	4693
1472	4473	4503	4538	4578	4623	4673	4729
1536	4516	4545	4579	4617	4661	4710	4765
1600	4556	4585	4618	4655	4698	4746	4800
1664	4596	4624	4656	4693	4735	4781	4834
1728	4635	4662	4693	4729	4770	4816	4867
1792	4673	4699	4730	4765	4805	4849	4899
1856	4710	4735	4765	4800	4839	4882	4931
1920	4746	4771	4800	4834	4872	4915	4963
1984	4781	4805	4834	4867	4904	4946	4994
2048	4815	4839	4867	4899	4936	4977	5024
2112	4849	4873	4900	4931	4967	5008	5053
2176	4882	4905	4932	4963	4998	5038	5082
2240	4914	4937	4963	4994	5028	5067	5111
2304	4946	4968	4994	5024	5058	5096	5139
2368	4977	4999	5024	5053	5087	5124	5167
2432	5008	5029	5054	5082	5115	5152	5194
2496	5037	5058	5083	5111	5143	5180	5221
2560	5067	5087	5111	5139	5171	5207	5247
2624	5096	5116	5140	5167	5198	5233	5273
2688	5124	5144	5167	5194	5225	5260	5299
2752	5152	5172	5194	5221	5251	5285	5324
2816	5179	5199	5221	5247	5277	5311	5349
2880	5206	5225	5247	5273	5302	5336	5373
2944	5233	5252	5273	5299	5328	5360	5397
3008	5259	5278	5299	5324	5352	5385	5421
3072	5285	5303	5324	5349	5377	5409	5445
3136	5310	5328	5349	5373	5401	5432	5468

Object and Ambient temperatures in deci-Kelvin [dK]. Pixel voltage in digits [dig]. Insert sensitivity (and emissivity) corrected voltage.

Table Number #11

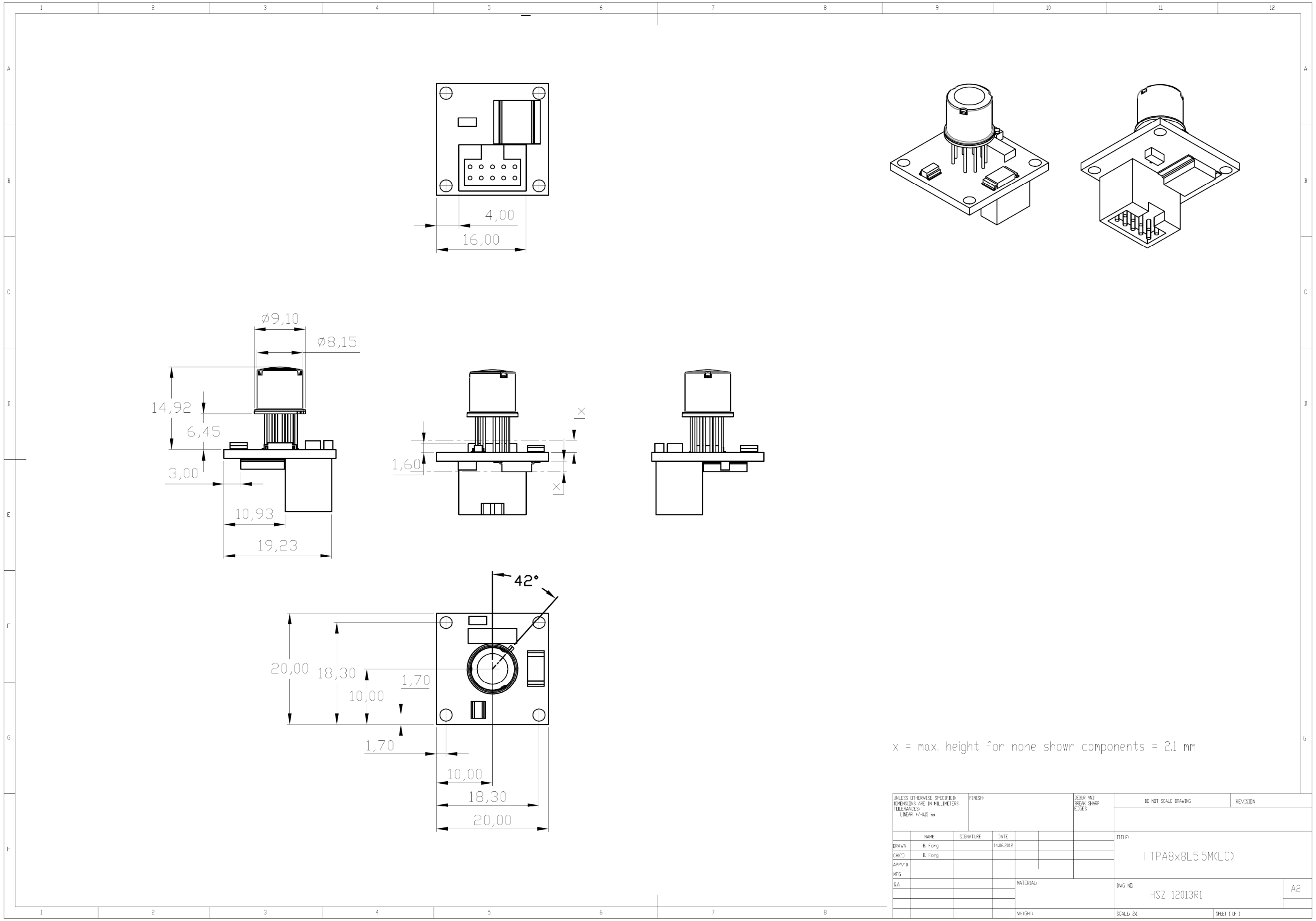
You can find the matching table number to your device in the EEPROM, refer to “EEPROM Mapping”

HEIMANN Sensor GmbH
Grenzstr. 22
D-01109 Dresden / Germany

Contact / Customer Support
Phone 49 (0) 6123 60 50 30
Fax 49 (0) 6123 60 50 39

Internet
www.heimannsensor.com
mail: info@heimannsensor.com

Dimensions:



HEIMANN Sensor GmbH
Grenzstr. 22
D-01109 Dresden / Germany

Contact / Customer Support
Phone 49 (0) 6123 60 50 30
Fax 49 (0) 6123 60 50 39

Internet
www.heimannsensor.com
mail: info@heimannsensor.com