

---

# m2Talk smart monitors

---

A smart monitor system that extends product lifetime, increases in-service operation without the need for additional wiring.

---

GPEG International Ltd

---

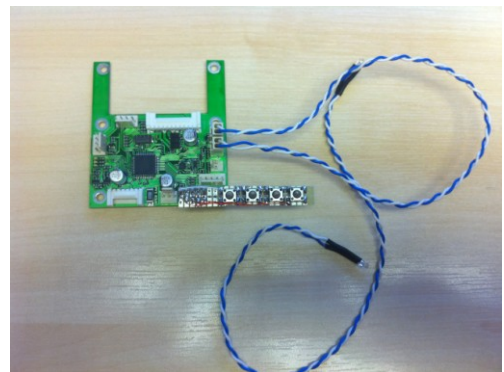


Figure 1 m2talk PCB

## Contents

Smart Monitors enabled by m2talk technology .....	3
Background .....	3
Features and Benefits Summary.....	3
Implementation .....	4
Hardware .....	4
Hardware Description - PCB Connectors: .....	5
Software .....	6
M2talk Features in detail .....	6
Monitor Identification.....	6
Shock and Tilt Sensors .....	6
Light and Temperature Sensors .....	6
Touch Screen Sensors .....	6
Power Sensors.....	7
Backlight Control and Sensing.....	7
Proximity sensors .....	7
Local Key switches.....	7
ccTalk simplifier/convertor .....	7
System background functions.....	8
Package contents .....	9
Installation .....	9
M2talk User Application .....	9
Appendix A – DLL description (v0.1) .....	10
Appendix B – PCB layout.....	12
Appendix C - Internal terminations.....	13
Appendix D – Light Sensor Specification.....	14
Appendix E – Proximity Detector .....	15

## Smart Monitors enabled by m2talk technology

### Background

For monitors used in revenue generating activities such as digital advertising, gaming and point of sale, the requirement for in-service up-time is of key importance.

In a typical installation the monitor is not easily accessible or serviceable, and often forms part of a large network of installations, across a wide geographic area.

In the past, network monitors have had diagnostic functions added to them through the use of expensive and somewhat complicated systems, requiring additional hardware, cabling and power.

M2talk enabled monitors overcome these limitations and deliver a wide range of benefits to anyone running a distributed set of monitors on either a local area (LAN) or wide area network (WAN).

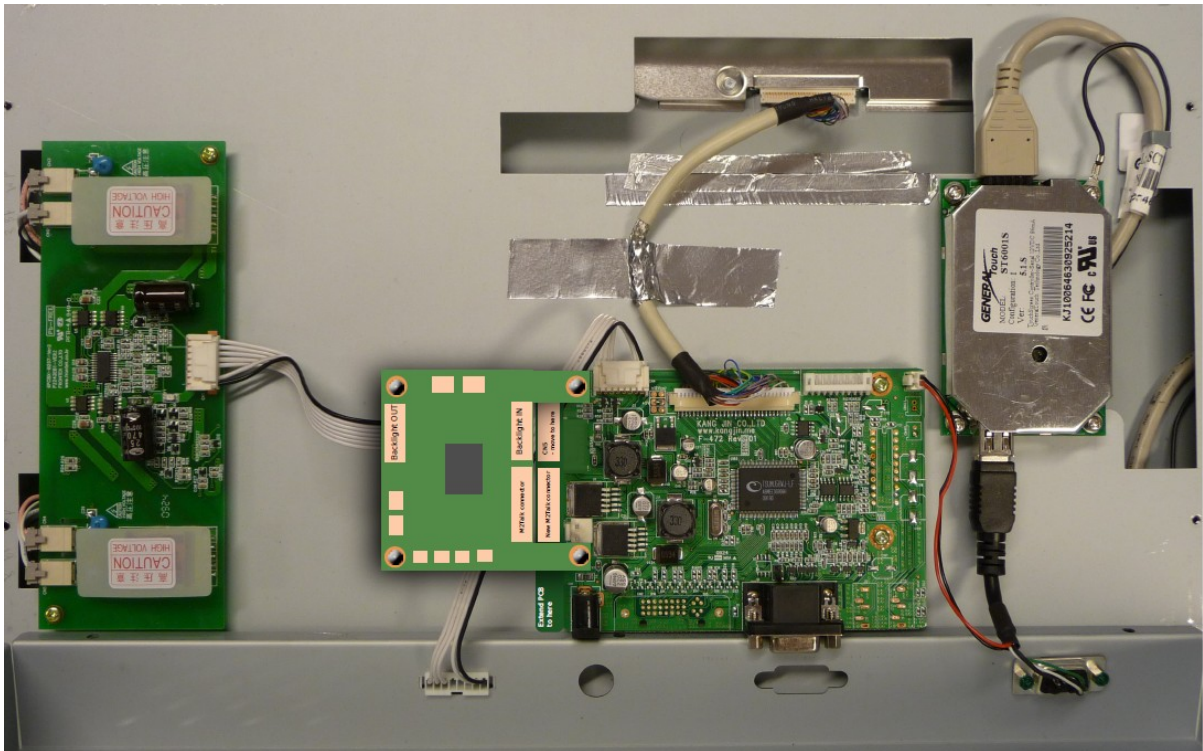
### Features and Benefits Summary

Feature	Benefit
Available sizes	9.4" to 104"
Input	VGA or DVI
Touch Decode	SAW touch
Multi Monitor	YES
Compatible Video Cards	nVidia, ATI, Intel, S3, Matrox, etc.
System Requirement	Windows 2000/XP/2003/Vista/Windows7
<b>Backlight Control and sensing</b>	Extended backlight lifetime, identify failures
<b>Shock and Tilt sensors</b>	Help protect your investment from damage
<b>Light and Temperature sensing</b>	Extend in-service lifetime, monitor environment
<b>Independent Touch Screen Sense</b>	Remotely check touch screen health
<b>Power sensing</b>	Create alerts for potential faulty components
<b>Proximity detection</b>	Assess and react to real events, and review audience patterns for improved revenue
<b>Monitor Identification</b>	Greater system awareness and integrity
<b>ccTalk simplifier</b>	Reduce wiring and need for serial ports
<b>Local keypad</b>	Add intelligent service and trained operator functions locally to the monitor
<b>Upgradeable FW</b>	In-field upgrades and customisation possible

## Implementation

### Hardware

M2talk exists as a daughter board that is connected internally to the main A/D card inside the monitor. It uses existing fixing holes and connectors so is simple and reliable to install both at manufacture and in a qualified service centre.



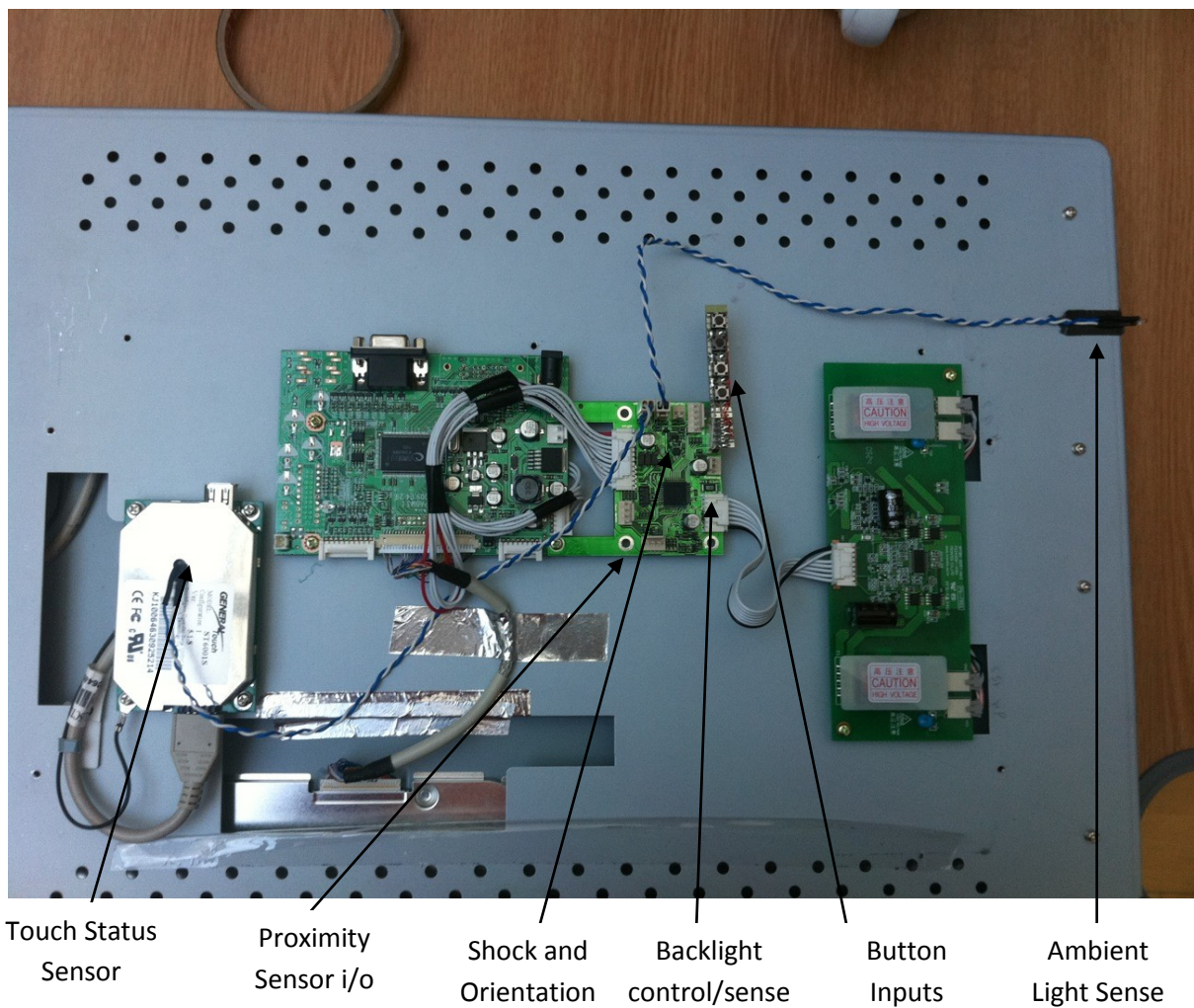
The hardware contains the following sensors:

- A/D board and VGA/DVI cable power sense
- Intelligent Backlight Power sense
- Smart Touch screen sense
- Shock, vibration and orientation sense
- Proximity sensor
- Temperature and ambient light sensor

The hardware fits easily into the existing metalwork and mates with one of several GPEG video cards and can be implemented on monitor sizes from 9.4" to 104". It is powered directly from these video cards or from the 5V VGA line (if available) but separately from the A/D board. This enables an m2talk system to operate even in the absence of internal power and retain the ability to post a fault event to the system.

## Hardware Description - PCB Connectors:

Function	Header	Function
DDC	J9	4 pin (Gnd, +5V, SCL, SDA)
BL	J9	2 pin (Gnd, BL)
Buttons	J8	5 pin (Gnd, B1, B2, B3, B4)
BL light sensor	J6	2 pin (Gnd, BLS)
Ext.light sensor	J2	2 pin (Gnd, ELS)
Temp sensor	Int	2 pin (thermistor)
Proximity sensor	J4	3 pin (Gnd, +5V, Prox), note the +5V from the A/D card,
ccTalk	J7	4 pin (Type 1: +5V, <key>, Gnd, -Data)
ISP	J10	5 pin (MCU programmer connection)



## Software

M2talk is provided with a set of DLL (dynamic link library) is provided which is placed on the host PC allowing polling and communication with the m2talk daughter board via the existing VGA or DVI header. Once installed the full range of facilities available on m2talk can be accessed and utilised either locally or via an existing network connection to and from the host PC.

It is intended that once the DLL is installed the network owner can develop a set of actions, based on the information provided by the m2talk monitor using the supplied software API. In this way intelligence can be overlaid on the monitor estate enhancing utility and improving services.

## M2talk Features in detail

### Monitor Identification

- DDC enumerating (monitors)
  - Multi-monitor setup can be easily identified
- Set m2talk i2c address
  - Flexible implementation
- Check for m2talk capable monitors
  - Auto sensing for compatible monitors for greater system integrity
- Get/Set serial number (=> EEPROM or factory fix)
  - Identify individual units anywhere in the field for service/diagnostic/lease support

### Shock and Tilt Sensors

- Get accelerometer (3 x 10bit data on 6 bytes) for determine orientation
  - Automatically use Window rotate feature on an enabled Monitor
  - Use as a highly reliable Tilt sensor to help prevent monitor abuse
- Set configure accelerometer (sensitivity, shock detect)
  - Adjust any implementation with “shock” parameters
  - Calibration of expected shock events to prevent abuse
- Poll accelerometer event (if enabled)

### Light and Temperature Sensors

- Get Ext. Light sensor (0-255)
  - Compact Light dependent transistor can be used to sense ambient lighting for backlight lifetime extension
- Get Temp sensor
  - Internal temperature sensor can be used to monitor environment and reduce backlight brightness for lifetime enhancement

### Touch Screen Sensors

- Get BL/Touch light sensor (0-255)
  - Backlight light sensor can be implemented internally to assess lighting condition (panel dependent)
  - Touch light sensor has decode for LED status used on GPEG SAW touch monitors and will report back the touch screen status

## Power Sensors

- Poll m2talk for A/D state (normal/damaged)
- Poll m2talk for backlight drive state (normal/damaged)
- Poll m2talk for missing VGA power (normal/damaged)

## Backlight Control and Sensing

- Get BL current
  - Assess backlight condition or malfunction
- Get Buttons
  - Local 4 key button array implemented for diagnostics and high level operation across the m2talk firmware
- Get/Set Backlight (0-100% or 0-255)
  - Remotely set and adjust backlight brightness in real-time
  - CFL adjusts 50~100% (typically), LED from 5~100%
- Configure monitoring PC sys-wide mouse and keyboard activity.
  - Automatically decrease backlight to a given percentage after inactive interval elapsed.

## Proximity sensors

- Get proximity sensor (0-255)
  - Local proximity sensor can be utilised for advertising/call to action as well as monitor/game condition adjustments
- Set proximity event parameters (to count/estimate users)
  - Traffic estimates can be gathered from the proximity sensors to help with maximising cabinet positioning within a building or store
  - Monitor traffic levels and feedback together with advertising data
- Poll proximity event (if enabled)

## Local Key switches

- Poll Button events
- Link events to user functions by utilising internal key decode routine

## ccTalk simplifier/convertor

- Gain and extra serial port by sending ccTalk data via the m2talk enabled monitor.
- Reduce wiring loom complexity
- Enable ccTalk devices localised with monitor remote from serial communications
- ccTalk:
  - Set Baud
  - Get MCU raw fifo
  - Clear MCU fifo
  - Send ccTalk command
  - Set timeout for response
  - Start Poll ccTalk response (set period)

- Stop Poll ccTalk response
- ...ccTalk commands and responses should be processed in user app

## System background functions

- Set password => EEPROM
- Get FW version
- Set MCU-alert-pin
- Set/enable/disable Autopoll-m2talk events timer
- Open config panel for setup all configurable things by user (uses systray icon)
- Timers for poll events, timeouts
- FW update:
  - Set start FW download, continue if acknowledged
  - Set erase current FW
  - Get busy
  - Set send 256 bytes + checksum (send the entire FW in 256 bytes packets)
  - Set FW update complete => starts the new FW

## Package contents

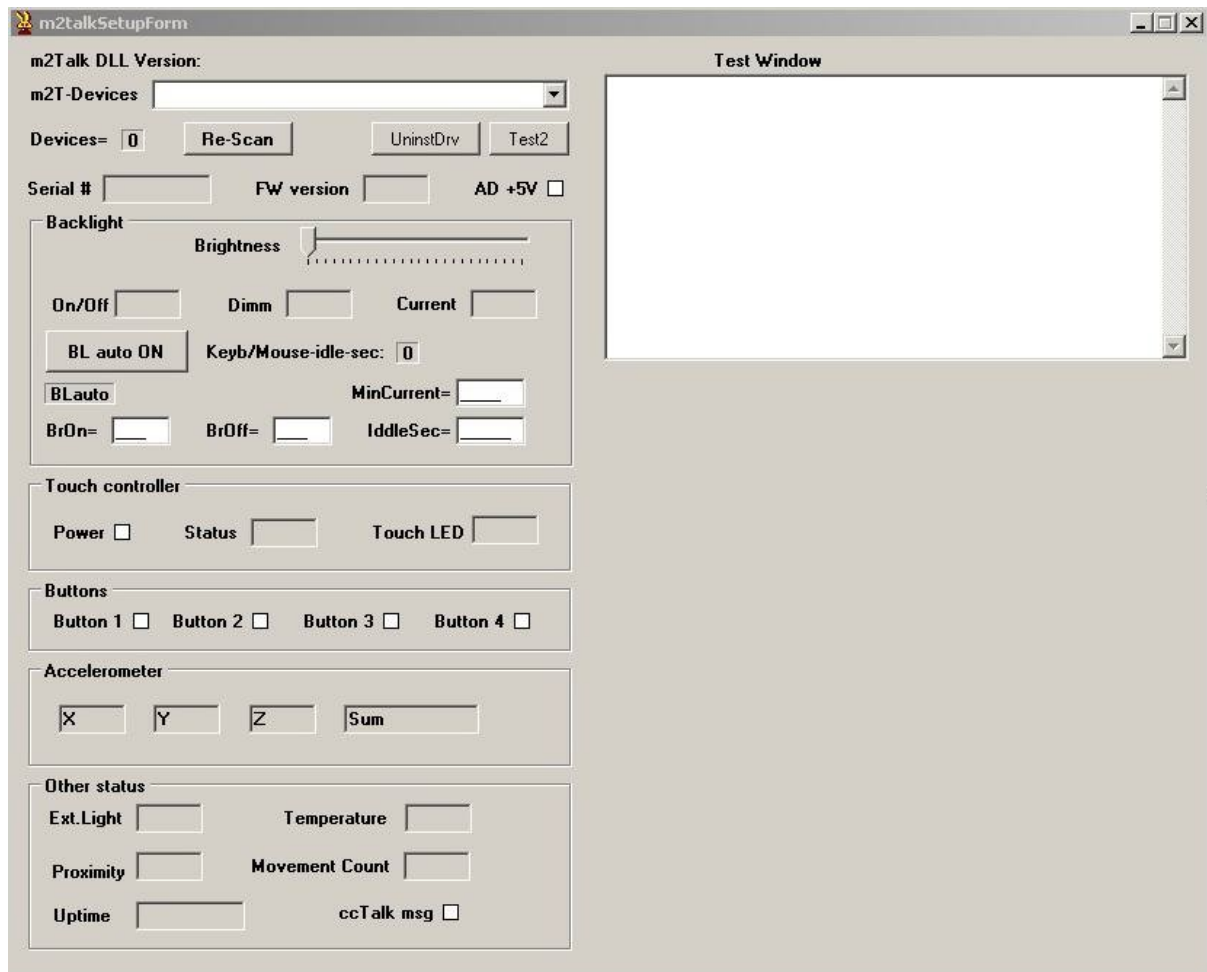
File	Type	Description
M2talk.DLL	Dynamic link library	
M2talkuserapp	Exe application	
M2tdrv.sys	Sys file	
M2t_DDC	Dynamic link library	

## Installation

To be defined

## M2talk User Application

To help familiarise the user with the features implemented on your m2talk monitor, the User application provides a consol that shows the data and functionality in real-time from your monitor. It can be used in development of your network application, to test and calibrate the units and as a demonstration tool.



## Appendix A – DLL description (v0.1)

/\*

m2Talk API.h: Usage of m2Talk.dll (c#)

Add reference to m2Talk.dll

All public constants and functions are in class m2talkDll.

All functions and variables are static.

At the startup, the user app first should call open\_m2talk().

It will install our driver ddcdrv.sys if necessary at the first run.

The user should have administrator privileges at this time.

If an error occurs, it displays message box about the error.

It will enumerate connected displays, verifies which is m2talk enabled.

\*/

public class m2talkDll:

// Public functions:

public static bool m2talkDll.open\_m2talk()

Must call at startup of the app.

ret: True if success, False if error.

-installs driver m2TDrv.sys if not installed, the app requires administrator privileges at this time.

-starts driver, enumerates m2Talk devices (multi monitor system),

-starts system hooks.

-fills the followings for user app:

// !!!!!!!! change for only m2T enumerated!!

public static int DevCnt;

// count of m2Talk devices. If 0: no devices found

public static string[] Dev\_names[10]; // display names (DDC EDID, assigned by manufacturer)

public static bool[] Dev\_m2t[10];

public static bool DDC\_OK;

// True if the m2Talk.dll can connect to the

DDC hardware

public static bool m2talk\_OK;

// True if one or more m2Talk device found

public static bool m2talkDll.close\_m2talk()

Must call before app exit. Closes driver and system hooks.

public static m2tResp set\_BackLight(int disp, byte intensity)

sets backlight BL-dimm value and BL-On/Off

disp: device index (0..DevCnt-1)

intensity: 0=Off & BrMin, 1-255=On & Brightness

ret: m2tResp

public static int getAddr() // ??? public?

ret: i2c address used by m2talk

public static int setAddr(byte addr) // ??? public?

sets i2c address used by m2talk.

ret: 0=OK, 1=err: address-change-disabled

public static m2tResp m2T\_StatRead(int disp)

The user app should call this periodically

disp: device index (0..DevCnt-1)

ret: m2tResp

reads all sensors and status data from the m2Talk board:

public static UInt32 Serial;

public static byte

FW\_major, // major index of m2Talk board FW version

FW\_minor, // minor index of m2Talk board FW version

STATUS1, // bit field, detailed below

STATUS2, // not used

STATUS3, // not used

STATUS4; // not used

BUTTONS, // bit field, uswed <0..3>, detailed below

BLdim, // BL dimm (brightness) value 0..255 = +5..0V control voltage

ProxCnt, // counts distance changes (treshold setup!) => movments, count peoples...

R\_BLsens, // Backlight current, 0..255 ~ 0-3.5A (~13-15mA/bit, depending on +5V)

R\_ExtLight, // External light sensor, 0..255, uncalibrated

R\_TouchLED, // Now light sensor, 0..255, uncalibrated. later preprocessed

touchscreen activity

R\_Prox;

// Proximity distance, 0..255, unit depends on the used sensor

public static char Temp; // Temperature of m2Talk board. Signed, -128..+127 deg Celsius

```

public static Int16 // Accelerometer data. Signed, 3D, the maximum since last
m2T_StatRead()
    AccX,
    AccY,
    AccZ;
public static UInt32 UpTime; // The uptime of the m2Talk board (BL?), hours, stored/summed infinitely in
EEPROM
public static bool // (STATUS1 bits parshed to bool)
    B1on, // Backlight inverter is ON
    TouchPwr, // Touchscreen power OK
    Pwr5V, // Display AD card +5V OK
    new_ccTalk; // New ccTalk response arrived
public static bool // (BUTTONS bits parshed to bool)
    B1, // BUTTONS<0>: B1 pressed since last m2T_StatRead()
    B2, // BUTTONS<1>: B2 pressed since last m2T_StatRead()
    B3, // BUTTONS<2>: B3 pressed since last m2T_StatRead()
    B4; // BUTTONS<3>: B4 pressed since last m2T_StatRead()

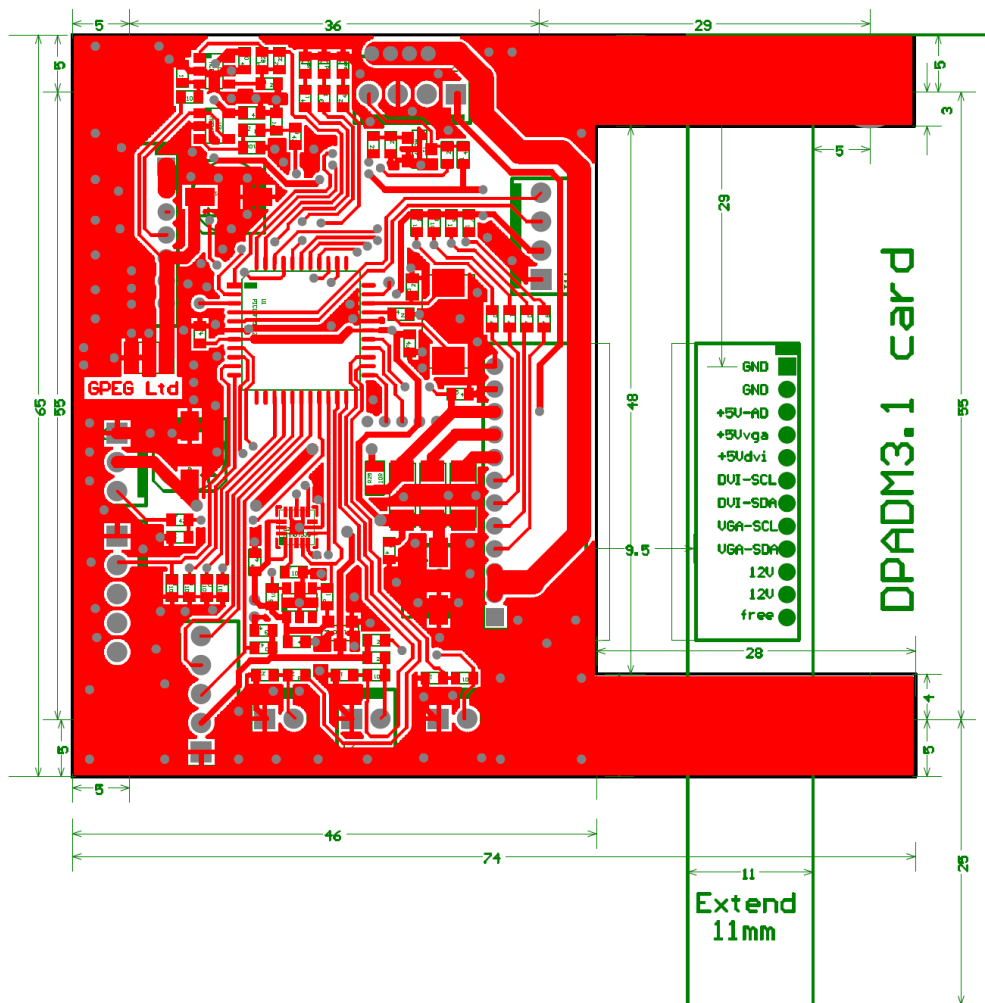
//Other public Variables:
public const string m2talkVersion = "1.0"; // version string of m2Talk.dll file
public static bool BL_auto = false; // enable/disable Backlight auto-On/Off control upon keyb/Mouse
idle time
public static byte BL_off_level = 50, // the brightness of BL-Off state
    BL_on_level = 255; // the brightness of BL-On state
public static UInt32 BL_km_sec_max = 20; // BL off after BL_km_sec_max sec key/mou idle ellapsed
public static UInt32 BL_km_idle_sec_cnt = 0; // count of key/mou idle sec. Resets on every key/mou events

public static int CurInd; // actual display, used only for test !!!!!

//constatnts:
//Error codes:
public enum m2tResp.....

```

## Appendix B - PCB layout



## Appendix C - Internal terminations

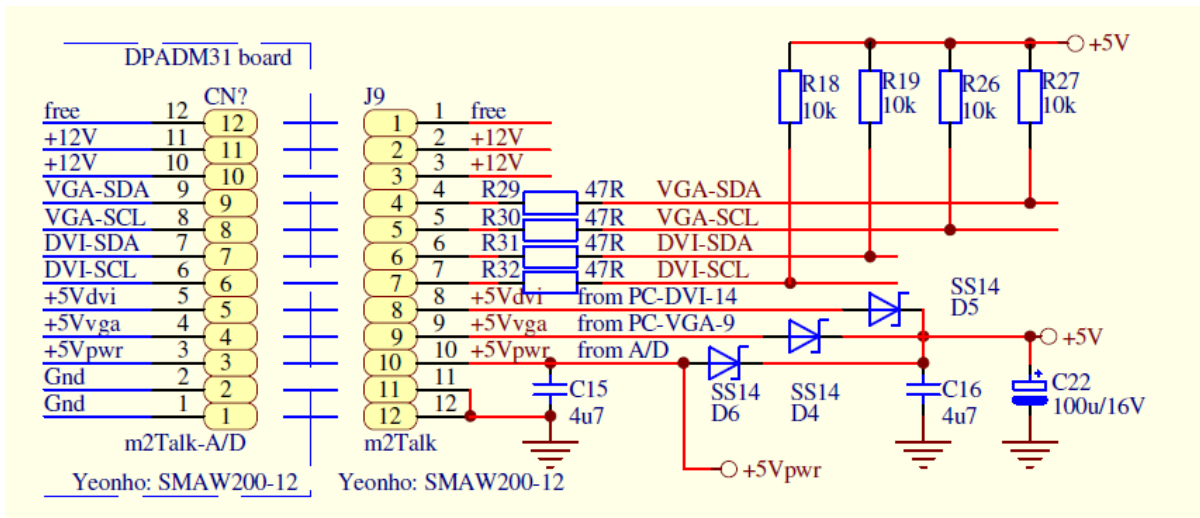


Figure 2 internal a/d card to m2talk board

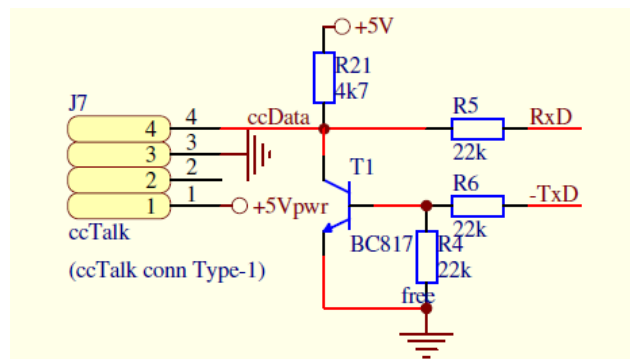


Figure 3 ccTalk internal termination

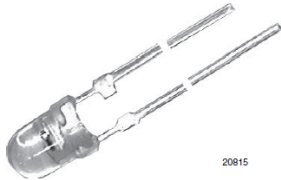
# Appendix D – Light Sensor Specification



**TEPT4400**

Vishay Semiconductors

## Ambient Light Sensor, RoHS Compliant



### FEATURES

- Package type: leaded
- Package form: T-1
- Dimensions (in mm): Ø 3
- High photo sensitivity
- Adapted to human eye responsivity
- Angle of half sensitivity:  $\varphi = \pm 30^\circ$
- Compliant to RoHS directive 2002/95/EC and in accordance to WEEE 2002/96/EC



**RoHS**  
COMPLIANT

### DESCRIPTION

TEPT4400 ambient light sensor is a silicon NPN epitaxial planar phototransistor in a T-1 package. It is sensitive to visible light much like the human eye and has peak sensitivity at 570 nm.

### APPLICATIONS

- Ambient light sensor for control of display backlight dimming in LCD displays and keypad backlighting of mobile devices and in industrial on/off-lighting operation
- Replacement of CdS photoresistors

### PRODUCT SUMMARY

COMPONENT	$I_{PCE}$ ( A)	$\varphi$ (deg)	$\lambda_{0.5}$ (nm)
TEPT4400	200	$\pm 30$	440 to 800

**Note**

Test condition see table "Basic Characteristics"

### ORDERING INFORMATION

ORDERING CODE	PACKAGING	REMARKS	PACKAGE FORM
TEPT4400	Bulk	MOQ: 5000 pcs, 5000 pcs/bulk. Label with $I_{PCE}$ group on each bulk. Specifications of group A/B/C see table "Type Dedicated Characteristics" on page 2	T-1

**Note**

MOQ: minimum order quantity

### ABSOLUTE MAXIMUM RATINGS

PARAMETER	TEST CONDITION	SYMBOL	VALUE	UNIT
Collector emitter voltage		$V_{CEO}$	6	V
Emitter collector voltage		$V_{ECO}$	1.5	V
Collector current		$I_C$	20	mA
Power dissipation	$T_{amb} \leq 55^\circ\text{C}$	$P_V$	100	mW
Junction temperature		$T_J$	100	$^\circ\text{C}$
Operating temperature range		$T_{amb}$	- 40 to + 85	$^\circ\text{C}$
Storage temperature range		$T_{stg}$	- 40 to + 100	$^\circ\text{C}$
Soldering temperature	$t \leq 3$ s	$T_{sd}$	260	$^\circ\text{C}$
Thermal resistance junction/ambient	J-STD-051, soldered on PCB	$R_{thJA}$	300	K/W

**Note**

$T_{amb} = 25^\circ\text{C}$ , unless otherwise specified



---

**About GPEG International Ltd.**

Based in London, with offices in Germany and Taiwan and manufacturing facilities in China and South Korea, GPEG is recognised as the fastest growing hardware business in the UK. With a group turnover in excess of \$35M we are focussed on delivering electronic solutions that are brighter, clearer and easier to use. GPEG helps customers in gaming, consumer, signage and other similar markets achieve faster time to market with more compelling products, supported by a variety of related patents and IP.