SCHURTER
ELECTRONIC COMPONENTS

User Manual

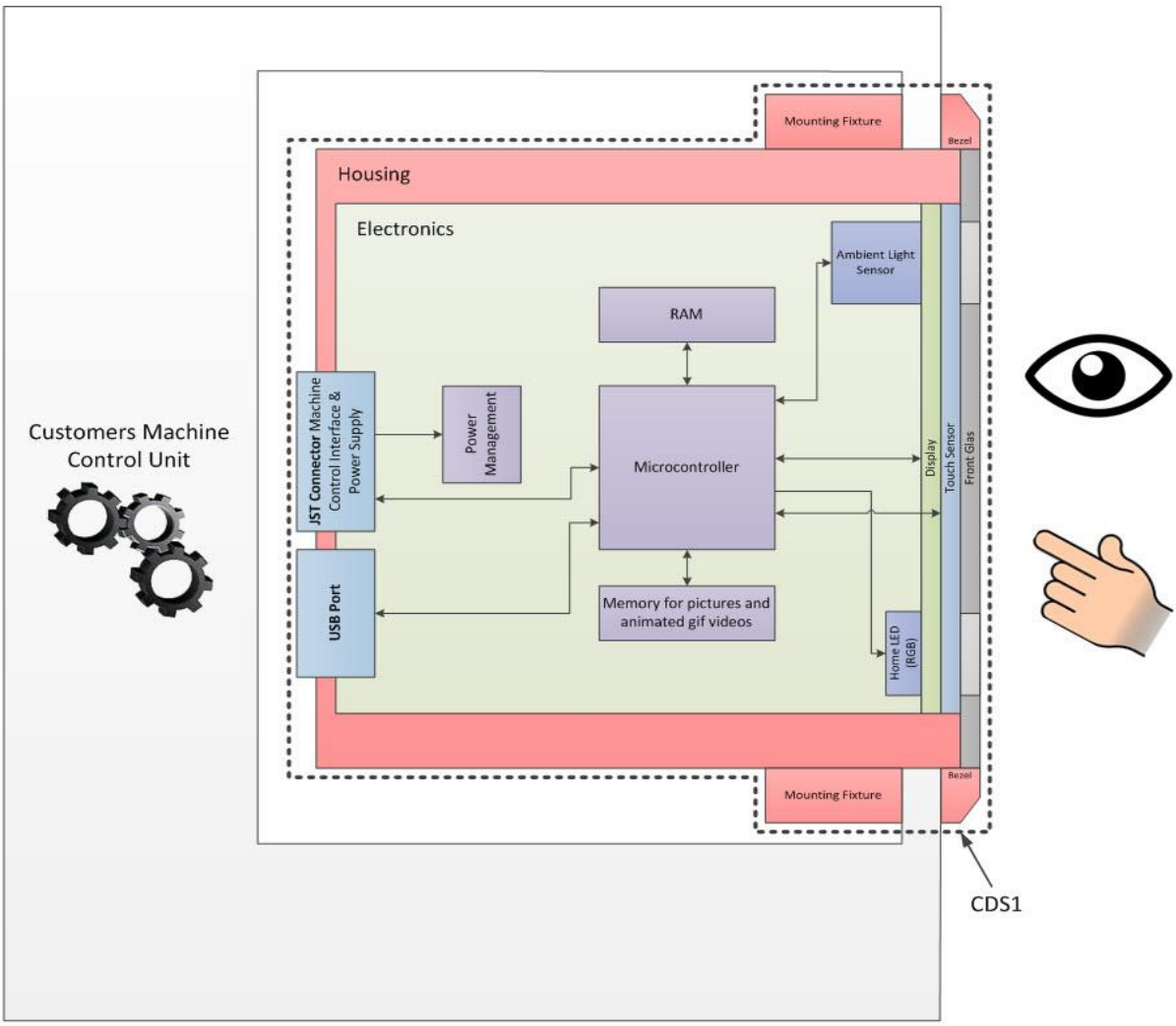
CDS1 Configurable Display Switch

Table of Contents

1 General product description

The CDS1 is a component that acts as a graphical interface between human and machine, also called HMI (human machine interface). It is a plug-n-play solution which can be easily implemented into a higher machine system.

The user input to the CDS1 is done via a capacitive touch area consisting of a touch display in the center on the CDS1 front side surrounded by a touch wheel area. The customer can assign functions to five soft keys. Four of these soft keys are located on the touch wheel area and one in the center of the display area. The input to these soft keys will be processed by a microcontroller. The microcontroller provides the resulting signals via the interface to the customers machine control unit which has to take care about the execution of the according instructions within the customers system. Three different interface protocols are available for the communication between the CDS1 and the machine control system. These are the SPI, I2C and RS232 interface.



The CDS1 has a built-in mass storage. This makes the CDS1 capable to store user defined pictures and animated gif videos. Thus the input for the operation of the overall system can be visually supported. The transfer of the user defined pictures and animated gif videos to the mass storage of the CDS1 is done via its Micro USB-B 2.0 port before the CDS1 is built in to its final application. For pictures only png format with a resolution of 128 x 128 pixels can be used. The addressing of the pictures and animated gif videos has to be done from the machine control system of the customer.

There are two touch input areas on the CDS1 based on PCAP technology. The first one is the outer touch wheel ring where the user can do his selection by a rotation movement on the touch wheel. In addition there are four soft keys located on this touch wheel.
The second touch input area is the so-called touch button in the center of the display. This touch button can be used to acknowledge a selection. As an alternative to the rotation on the touch wheel the user can swipe either horizontally or vertically over the display to scroll through a set of pictures.

2 Mounting instructions

The CDS1 Design-In Kit includes the following mounting accessories:
- Mounting ring
- O-Ring
- Screws

The CDS1 can be mounted to a front panel from the backside. It can be done either by using a mounting ring which will be glued to the customers machine housing, or the customer can prepare his machine housing with integrated screw domes. In the first case, the thickness of the panel can be between 1 mm and 6 mm.

Assembly steps to mount the CDS1 to a front panel using the mounting ring:

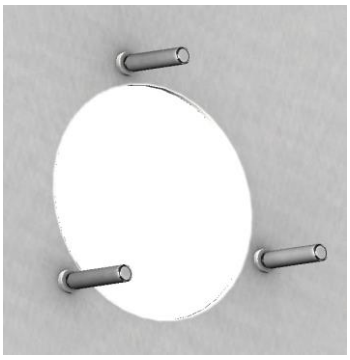
1 Prepare the mounting ring:
No special actions are required in case the front panel thickness is 1 mm. If the front panel thickness is more than 1 mm, the screw domes of the mounting ring need to shorted accordingly to get the CDS1 front aligned with the front panel surface. The mounting ring screw domes can be adjusted according your needs.

2 Attach the mounting ring:
Remove the protective foil from the adhesive tape on the mounting ring and glue the mounting ring to the front panel. Take care on the right orientation and perfect alignment, see the picture below.

3 Attach the CDS1 to the mounting ring.

4 Fix the CDS1 with the screws.

Alternatively, the screw domes could be integrated into the customers front panel housing, i.e. directly molded into a plastic panel as shown in the pictures below.



The CDS1 can either be mounted shortly to the front panel or protruding in case of using a front bezel.

3 Connecting the CDS1 for using the machine simulator on your PC

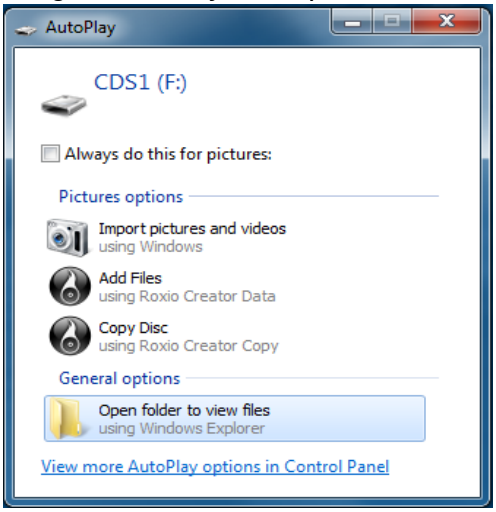
The Design-In Kit includes a machine simulator software which is stored on the CDS1. Alternatively the latest version of this software can be downloaded from the CDS1 landing page:
cds1.schurter.com

To prepare your CDS1 device, you need to follow the description below step by step.

1. Connect the JST connector to the CDS1 and plug the power adapter into an electrical socket. Check if the Home Button LED is blue. After the initialization phase, the Home Button LED switches off and the USB mass storage is available now.

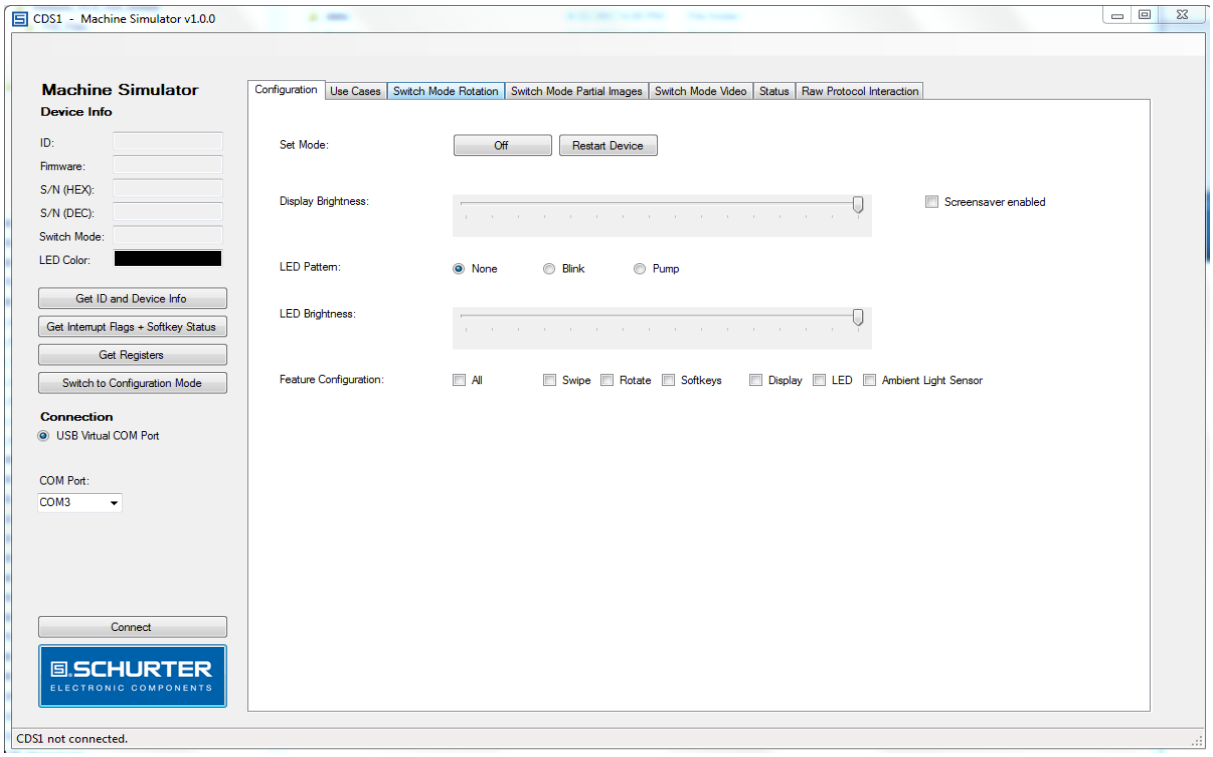
2. Connect the USB cable to the CDS1 and to a USB port of your PC. Make sure that the CDS1 is recognized by your PC (check the device manager). The recognition of the CDS1 may take some time. The more files are stored on the CDS1, the longer it may take. For some windows operating systems it might be necessary to install a USB driver to get the CDS1 device recognized as a USB device. The USB driver software can be downloaded from the SCHURTER CDS1 landing page.
   cds1.schurter.com

3 Open the mass storage CDS1 in your Explorer.



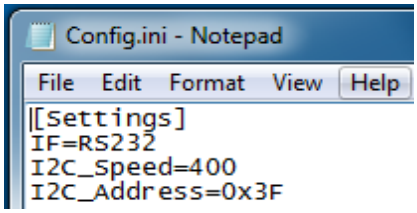
4 Copy the folder "Machine_Simulator" to your desktop.

5 Open the directory "Machine_Simulator" on your desktop and start the Machine Simulator by executing the file "CDS1_MachineSimulator.exe". This will start the Machine Simulator in a new window.

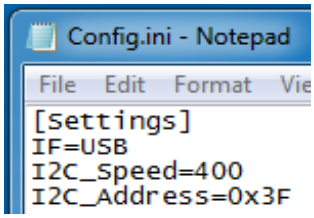


6 Open the Config.ini file **on the mass storage of the CDS1**, change the interface from RS232 to USB and save this change to the Config.ini file.

before change:



after change:

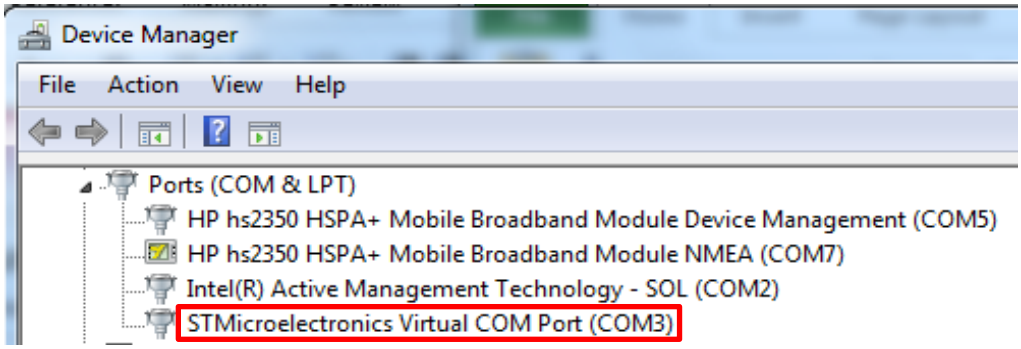


Attention:
After a new power-up, the CDS1 will start in USB interface mode. This means, that the mass storage of the CDS1 will not directly open after the power-up.

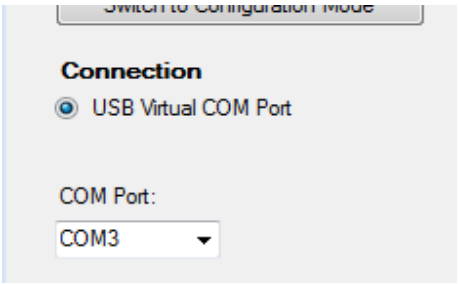
7 Make a power cycle by disconnecting and reconnecting the power supply of the CDS1.

8 Wait until the CDS1 is recognized as USB device (check in the device manager, see step 3).

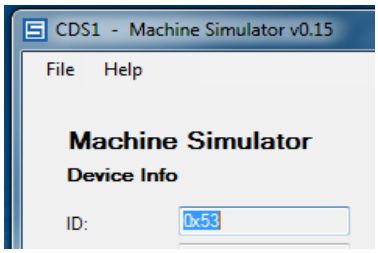
9 Check in the Device Manager which COM port is assigned to the STMicroelectronics Virtual COM Port

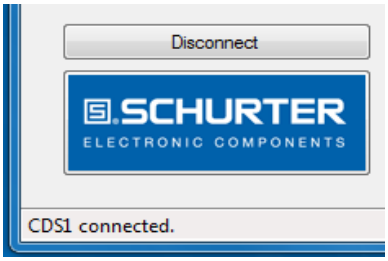


10 Set the COM port in the Machine Simulator to the port on which the CDS1 device is recognized in the device manager.



11 Push the "Connect" button to connect the CDS1. The button will now change to "Disconnect" button. In the footer line you can now see that the CDS1 is connected. In addition, the ID code of the CDS1 is shown on the left side of the window.





12 Define in the section "Feature Configuration" the features that you want to use:

13 Now you can start your work with the CDS1.

4 The Machine Simulator Software

System commands

Get ID and Device Info	Read out the Identification number and the Firmware Version.
Get Interrupt Flags + Soft key Status	Get the status of the interrupt register and the soft key registers. The contents of these registers can be seen in the functions tab "Status". If the contents of a register has changed its value i.e. due to an activated soft key, it is marked with yellow background color.
Get Registers	Read out of all registers from the CDS1. The contents of all registers can be seen in the functions tab "Status". Registers with a changed value are marked with yellow background color.
Switch to Configuration Mode	The mass storage is enabled and accessible via the USB connection to the PC. In this mode the customer can now save new or delete unwanted pictures or animated gif videos on the CDS1.

! Attention:
The set of pictures and animated gif videos must be identical on the CDS1 and in the "images" directory below the directory "Machine_Simulator" on the desktop. It is important to see the same pictures and animated gif videos in the Machine Simulator software as the pictures which are shown on the CDS1. In case of mismatches, the CDS1 will show an error message and the Machine Simulator software needs to be restarted.

Functions Tabs

Configuration	Define which of the switch features should be enabled, set the brightness of the display as well as the brightness and pattern for the Home Button LED.
Use Cases	Two use cases are available. A use case cannot be interrupted by another command from the Machine Simulator. The user must finish a use case to its end. Use case Touch Input: The user is requested to activate a sequence of different touch inputs. Use case LED: The user can choose the LED color and its brightness. The chosen configuration is directly visible on the Home Button LED.

	! Attention: While running a use case, the Machine Simulator Software is waiting for input to the CDS1. During this time, the Machine Simulator does not accept any further inputs via the graphical interface until the use case is finished. Use cases cannot be interrupted.
Switch Mode Rotation	Select the rotation mode, speed and sensitivity and transfer the chosen configuration to the CDS1 by pressing „Program Device“.
Switch Mode Partial Images	Select a background picture, enable the tick boxes for the partial pictures that should be shown and make your choice on which partial picture should be shown. For each enabled partial picture define the location of the partial picture on the display area by setting the x- and y-position value (origin is the left upper edge of the display area). Finally, transfer your settings to the CDS1 by pressing the „Program Device“ button.
Switch Mode Video	Select a video and decide whether it should run one time or in endless mode and start the video by pressing the „Play Video“ button.
Status	On this tab the status of all registers can be checked. Press „Get Registers“ to update the values.
Raw Protocol Interaction	The communication protocol between the CDS1 and the Machine Simulator is logged in this tab.

5 Interfaces

The communication between the CDS1 and the control unit of the machine can either via I2C (Standard or Fast Mode), Motorola 4-line SPI, RS232, or USB interface. The I2C, SPI and the RS232 interface use the JST connector, while for the communication via USB the USB connector is used.

To detect transmission errors the interface protocol includes a CRC-8 checksum which is built up as follows:

CRC Check sum	
CRC-8 Type	CRC-8-CCITT
Size	8 Bit
Polynomial	0x07
Initial	0x00

5.1 Pinning of the JST connector

JST XHP-10 connector pinout (interface to the machine control)				
Pin	**Type**	**SPI**	**I2C**	**RS232**
1	supply	VCC [1)]	VCC [1)]	VCC [1)]
2	GND	GND	GND	GND
3	output	IRQ_n [3), 4)]	IRQ_n [3), 4)]	IRQ_n [3), 4)]
4	input	CS_n [3)]	-	-
5	GND	GND	GND	GND
6	input	SCLK	SCL [2)]	-
7	GND	GND	GND	GND
8	input	MOSI	SDA [2)]	TX
9	GND	GND	GND	GND
10	output	MISO	-	RX

1) The power supply VCC has to be 3.3 VDC ±5% for all interface protocols. The minimum and maximum inout signal high and low levels are 70% VCC and 30% VCC.
2) Use external pull-up resistor, recommended value: 2.7 kΩ
3) Signal is active low
4) IRQ_n is a signal additionally to the serial interfaces which is internally connected to logic high level via an internal pull-up resistor. This signal will be tied to low level as soon as at least one interrupt flag is set.

5.2 Pinning of the USB port

USB port pinning			
Pin	**Type**	**Name**	**Comment**
1	supply	VCC	not connected
2	inout	D-	negative differential data line
3	inout	D+	positive differential data line
4		ID	not connected
5	GND	GND	ground

The CDS1 will be recognized as mass storage via the USB port. The mass storage contains the configuration file as well as the pictures and videos.

The USB port can be used for the communication between the CDS1 and the machine simulator or as CDC (Communication Device Class) which allows to use it as RS232 interface where the USB acts as a virtual Com Port. This is useful for program debugging when a customer machine control unit is not yet available. The interface protocol is the same as for the RS232 interface. Nevertheless, it is not recommended to use the USB interface as a permanent interface connection between the CDS1 and the customers machine control unit. In addition the power supply to the CDS1 is not done via the USB port.

5.3 Interface protocol selection

The customer can choose to use either between the I2C, SPI or RS232 interface protocol. The selection for one of these interfaces has to be done in the Config.ini file in the root directory on the CDS1. For all interfaces the input logic low level is 30% VCC and the input logic high level is 70% VCC.

config.ini file content

Line	Parameter	Options	Description
1	[Settings]	-	Do not change.
2	IF=	I2C	Set the interface to I2C interface.
		SPI	Set the interface to 4-line SPI interface. Line 2 and 3 do not have any meaning when the SPI interface is selected.
		RS232	Set the interface to RS232 interface. Line 2 and 3 do not have any meaning when the SPI interface is selected.
		USB	Set the interface to USB interface. When the interface is set to USB mode, the USB interface is used for the communication between the CDS1 and the machine simulator. In this case the USB acts as virtual COM port and therefore, the mass storage of the CDS1 is not accessible after a restart of the device. To get access again to the mass storage, the device has to be set into the configuration mode via the button "Switch to Configuration Mode". in the machine simulator.
3	I2C_Speed=	100	The I2C interface speed is set to 100 kHz (Standard mode).
		400	The I2C interface speed is set to 400 kHz (Fast mode).
4	I2C_Address=	0x3F	Set the I2C slave address. The default I2C slave address is 0x3F.
5	IRQ_Packet=	ON	For RS-232 or USB configuration, an interrupt is to be carried out by sending an interrupt packet in addition to the signaling via the interrupt line. This is not possible for the I2C or SPI interface.
		OFF	Interrupt signal is only available via the interrupt line.

In case the Config.ini file is missing, RS232 will be used as standard interface.

5.4 SPI Interface

The switch is configured as slave, the machine control unit acts as master. The data transmission is controlled by the active low chip select signal. Data are triggered on the positive clock edge (SPI mode 0).

max. Baud rate: 1 MHz
Frame size: 8 Bit
Frame format: MSB first
SPI Mode: Mode 0

SPI write cycle:
The machine control unit pulls the chip select signal to logic low level and sends the write command bit together with the number of expected data bytes from the CDS1 (max. 127 bytes). Then the address of the first data register has to be sent followed by the related data. After every byte the register address is incremented automatically. After the last data byte was sent, the machine control unit must send a CRC-8 check sum which was calculated across the whole package and send it to the CDS1. If the transmission was successful, the CDS1 will acknowledge it by sending a write command together with the number of bytes sent. The acknowledge can be requested earliest 10 ms after end of the writing command. After this sequence the chip select is released by the machine control unit and the CS signal will be pulled up to high level.

SPI read cycle:
The machine control unit pulls the chip select signal to logic low level and sends the read command bit together with the number of expected data bytes to be read (max. 127 bytes).
Then the first register address and the CRC-8 check sum will be sent. The CDS1 answers with a read bit together with the number of bytes to be sent, followed by the read bytes. The register addresses increments automatically after each byte which was read.
After the last data byte was sent, the CDS1 sends a CRC-8 check sum (calculated across the whole package) to the machine control unit. After receiving the last expected byte, the machine control unit releases the chip select signal. Between a read command and a write command, a delay of minimum 10 ms is required.

SPI write command sequence

MOSI								MISO
write command + length	register addr.	byte1	byte 2	…	byte n	CRC-8	10 ms delay	write command + length

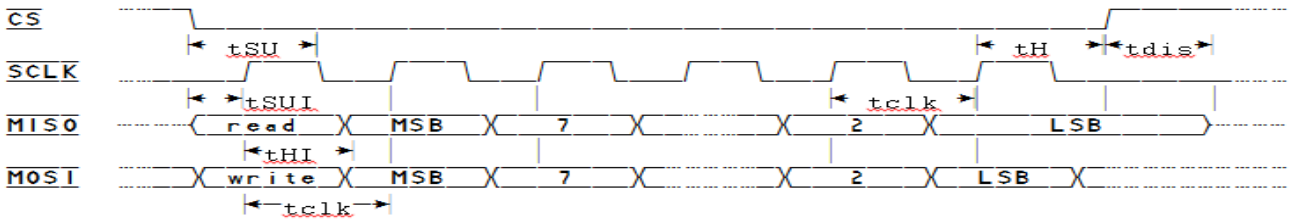
SPI read command sequence

MOSI				MISO					
read command + length	register addr.	CRC-8	10 ms delay	read command + length	byte1	byte 2	…	byte n	CRC-8

Command byte

Bit	Symbol	Value	Description
7	R/W		data read or write selection
		0	write data
		1	read data
6 to 0	length	0x00 to 0x7F	number of data bytes to write or to read

SPI timing diagram



tSU	chip select setup time	min. 4x tclk = 4 µs @ 1 MHz
tH	chip select hold time	min. 2x tclk = 2 µs @ 1 MHz
tSUI	data input setup time	min. 2.5 ns
tHI	data input hold time	min. 4 ns
tdis	data output disable time	max. 16.5ns
tclk	clock period	max. 1 µs @ 1 MHz

5.5 I2C interface

The CDS1 is configured as slave while the machine control unit acts as master.
The default I2C slave address of the CDS1 is 0x3F. This slave address and the I2C interface speed can be defined in the Config.ini file.

I2C write cycle:
The machine control unit sends the start condition and afterwards the write command consisting of the I2C slave address of the CDS1 and the R/W_n bit set for write, followed by a data byte which defines the number of bytes to be written. Then the address of the first instruction register needs to be sent followed by the data bytes. Each single byte will be acknowledged by an acknowledge bit. In addition, the register address is automatically incremented after each data byte and its acknowledge bit.
Once the last data byte was sent, the machine control unit has to send a CRC-8 check sum to the CSD1 which was calculated about the whole package. The CDS1 compares the preserved check sum to the calculated one. If a CRC error is detected, the CDS1 generates an interrupt. When the write sequence is finished, the machine control unit sends the I2C stop condition.

I2C read cycle:
The machine control unit sends the start condition and afterwards the write command consisting of the I2C slave address of the CDS1 and the R/W_n bit set to logic low for write. The next data byte contains the number of the bytes to be read. Then the address of the register address of the first register to be read is sent. A CRC-8 check sum follows and after the acknowledge bit from the CDS1, the I2C stop condition has to be sent.
Then the I2C start condition needs to be sent again, this time followed by the read command (I2C slave address with R/W_n bit set to 1) for starting the reading sequence. The data are read byte by byte, while the register address is incremented automatically after each byte read. As soon as all defined registers are read, the CDS1 will send a CRC-8 check sum. This check sum byte will be followed by a not-acknowledge bit. At the end of the read cycle, the machine control unit has to send the I2C stop condition to release the interface for new requests. Between the read command from the master and the start of clocking out the values from the slave a delay of at least 10ms must be inserted.

Slave address for the CDS1

Name	Slave address							
Bit	7	6	5	4	3	2	1	0
	I2C slave address							R/W_n

Bit	Symbol	Value	Description
7 to 1	I2C slave address	0x3F	slave address for the CDS1
0	R/W_n	0	write
		1	read

I2C write command sequence:

start	slave addr. + write	ack.	data length	ack.	register addr.	ack.	byte 1	ack.	…	byte n	ack.	CRC	ack.	stop

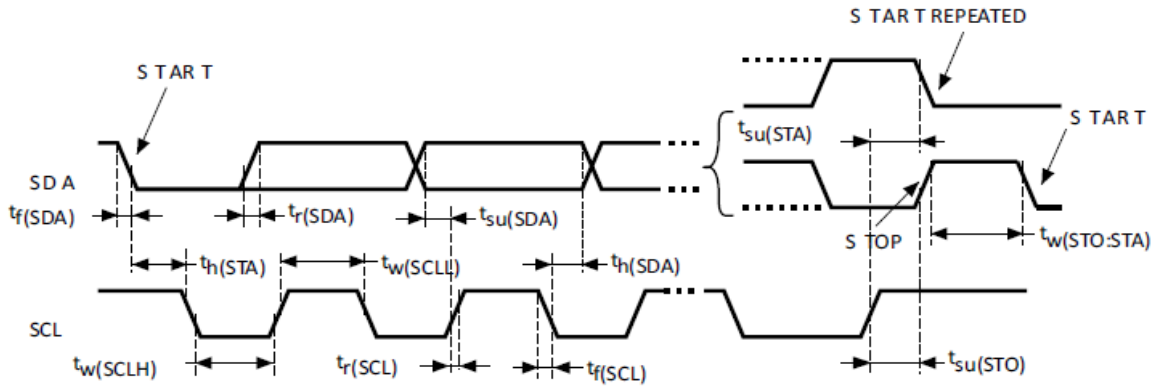
I2C read command sequence:

start	slave addr. + write	ack.	data length	ack.	register addr.	ack.	CRC	ack.	stop	10 ms delay

start	slave addr. + read	ack.	byte 1	ack.	…	byte n	ack.	CRC	no ack.	stop

from master to slave | from slave to master

I2C timing diagram



tw(SCLL)	SCL clock low time	100 kHz: min. 4.7 µs, 400 kHz: min. 1.3 µs
tw(SCLH)	SCL clock high time	100 kHz: min. 4.0 µs, 400 kHz: min. 0.6 µs
tsu(SDA)	SDA setup time	100 kHz: min. 250 ns, 400 kHz: min. 100 ns
th(SDA)	SDA hold time	0 ns
tr	rise time	100 kHz: max. 1.0 µs, 400 kHz: max. 300 ns
tf	fall time	100 kHz: max. 300 ns, 400 kHz: max. 300 ns
tsu(STA)	repeated start condition setup time	100 kHz: min. 4.0 µs, 400 kHz: min. 0.6 µs
th(STO)	start condition hold time	100 kHz: min. 4.7 µs, 400 kHz: min. 0.6 µs
tsu(STO)	stop condition setup time	100 kHz: min. 4.0 µs, 400 kHz: min. 0.6 µs
tw(STO:STA)	stop to start condition time	100 kHz: min. 4.0 µs, 400 kHz: min. 1.3 µs

5.6 RS232 interface

The RS-232 interface has a fixed Baud rate of 115'200 Baud and can therefore be used for cable lengths up to 2 m. The RS-232 mode is 8:N:1 which means 8 data bits, no parity bit and 1 stop bit.

RS232 write cycle:
The machine control unit begins of each data byte with a start bit and finishes the byte transfer with a stop bit. At the beginning the machine control unit sends the write command (1 bit) and the number of bytes to be written (7 bits, therefore maximum 127 bytes can be written). The next byte contains the first register address followed by the data bytes.
After the last data byte was sent, the machine control unit has to calculate the CRC-8 check sum across whole package and send it to the CDS1 device. If the transmission was successful, the CDS1 will acknowledge it by sending a write command together with the number of received data bytes followed by a CRC-8 checksum. In case the transmission was not successful, an interrupt will be generated.

RS232 read cycle:
The ready cycle begins with a read bit (1 bit) and the number of the bytes to be read (7 bits).
The next byte contains the first register address to be read. The read command ends with a CRC-8 check sum. The CDS1 answers with the read command and the number of data bytes to be read. Then the CDS1 sends the register start address and starts sending data bytes. The register address is incremented automatically after each sent byte.
After the last sent data byte the CDS1 will send a CRC-8 check sum. In case of a CRC error the machine should reject the data and request the data again.

RS232 Interrupt packet:
As soon as one of the Interrupt-Flags is set in the register INTERRUPT_FLAGS and if in the Config.ini file IRQ_Type is set to IRQ_PACKET, the Interrupt package is dispatched to the machine. The package consists as follows:

Format	10 bit	10 bit	10 bit
Content	**read / length**	**register addr.**	**INTERRUPT_FLAGS**
Example	00000 0001	0x02	0x03

The packet is sent by the CDS1 as a read command with the length 1 and the address of the register INTERRUPT_FLAGS (0x02) and the register contents. These data will be sent to the machine control unit. Above shown example indicates a Touch Button Interrupt.

6 Description of the registers

6.1 Register overview

Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	reset value
0x00	ID	ID CDS1								0x53
0x01	FW_VERSION	major version				minor version				
0x02	INTERRUPT_FLAGS	Reset/ Error	CRC Error	reserved	VID	Scroll	SK	TB	sum	0x81
0x03	BUTTON_STATUS	reserved						DBL	DBS	0x00
0x04	SOFTKEY_STATUS	RIGHT_L	TOP_L	LEFT_L	BOTTOM_L	RIGHT_S	TOP_S	LEFT_S	BOTTOM_S	0x00
0x05	MENU_STATUS	current picture (LSByte)								0x00
0x06		current rotation group (MSByte)								0x00
0x07	ALS_STATUS	ambient light value (LSByte)								0x00
0x08		ambient light value (MSByte)								0x00
…	…	…	…	…	…	…	…	…	…	…
0x10	FEATURE_CFG	reserved		swipe	wheel	SK	ALS	LED	display	0x00
0x11	IRQ_CFG	reserved	Reset / Error	CRC	reserved	VID	Scroll	SK	TB	0x6F
0x12	DISPLAY_CFG	reserved				brightness				0x0F
0x13	LED_CFG	reserved		pattern		brightness				0x0F
0x14	LED_COLOR_R	intensity red								0x00
0x15	LED_COLOR_G	intensity green								0x00
0x16	LED_COLOR_B	intensity blue								0x00
0x17	TOUCH_CFG	reserved				snap scroll sensitivity		rotation speed		0x0B
…	…	…	…	…	…	…	…	…	…	…
0x20	SWITCH_MODE	screen saver	mode parameter			reserved	config mode	mode		0x00
0x21	ROTATION	start picture number (LSByte)								0x00
0x22		rotation group (MSByte)								0x00
0x23	BACKGROUND	picture number (LSByte)								0x00
0x24		reserved (MSByte)								0x00
0x25	PART1	number of the first partial image (LSByte)								0x00
0x26		reserved (MSByte)								0x00
0x27	PART1 POSITION	reserved	X position of partial image number 1 (0 - 127) starting point is in the upper left edge of the display							0x00
0x28		reserved	Y position of partial image number 1 (0 - 127) starting point is in the upper left edge of the display							0x00
0x29	PART2	number of the second partial image (LSByte)								0x00
0x2A		reserved (MSByte)								0x00
0x2B	PART2 POSITION	reserved	X position of partial image number 2 (0 - 127) starting point is in the upper left edge of the display							0x00
0x2C		reserved	Y position of partial image number 2 (0 - 127) starting point is in the upper left edge of the display							0x00
0x2D	PART3	number of the third partial image (LSByte)								0x00
0x2E		reserved (MSByte)								0x00

Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	reset value
0x2F	PART3 POSITION	reserved	X position of partial image number 3 (0 - 127) starting point is in the upper left edge of the display							0x00
0x30		reserved	Y position of partial image number 3 (0 - 127) starting point is in the upper left edge of the display							0x00
0x31	PART4	number of the fourth partial image (LSByte)								0x00
0x32		reserved (MSByte)								0x00
0x33	PART4 POSITION	reserved	X position of partial image number 4 (0 - 127) starting point is in the upper left edge of the display							0x00
0x34		reserved	Y position of partial image number 4 (0 - 127) starting point is in the upper left edge of the display							0x00
0x35	VIDEO	video number (LSByte)								0x00
0x36		reserved (MSByte)								0x00
...	...	...	...	...	...	...	...	...	...	...
0xF6	TOUCH_STATUS	last pressed position on the touch wheel								0x00
0xF7	ERROR	Error Code								0x00
0xF8	RESET_REASON	LPWR RSTF	WWDG RSTF	IWDG RSTF	SFT RSTF	POR RSTF	PIN RSTF	BORRS TF	-	0x00
0xFA	DEVICE_ACTIONS	reserved							RESET	0x00
...	...	...	...	...	...	...	...	...	...	...
0xFC	SERIAL_NR	Serial Number Byte 0								-
0xFD		Serial Number Byte 1								-
0xFE		Serial Number Byte 2								-
0xFF		Serial Number Byte 3								-

6.2 General status and configuration registers

6.2.1 Readout the ID for the CDS1

The register ID contains an identification number which allows to identify the product as CDS1. This register is read only.

ID

Name	ID							
Addr	0x00							
Bit	7	6	5	4	3	2	1	0
r	ID CDS1							
w	-							
Reset	0x53							

Bit	Symbol	Value	Description
7 to 0	ID CDS1	0x53	Unique identification number of the LCD Switch

6.2.2 Readout the Firmware version

This register contains the actual installed version of the firmware. This value is read out when the CDS1 is powered up. The FW_VERSION register is read only.

FW_VERSION

Name	FW_VERSION							
Addr	0x01							
Bit	7	6	5	4	3	2	1	0
r	major version				minor version			
w	-				-			
Reset	firmware version							

Bit	Symbol	Value	Description
7 to 4	major version	0x0 to 0xF	Contains the pre-decimal places of the Firmware Release Version (i.e. 1 for 1.04)
3 to 0	minor version	0x0 to 0xF	Contains the decimal places of the Firmware Release Version (i.e. 4 for 1.04)

6.2.3 Configuration of the CDS1 features

FEATURE_CFG

Name	FEATURE_CFG							
Addr	0x10							
Bit	7	6	5	4	3	2	1	0
r	reserved		swipe	wheel	SK	ALS	LED	display
w								
Reset	0x00							

Bit	Symbol	Value	Description
7 to 6	reserved	00	Do not use
5	swipe	0	swipe feature is disabled
		1	swipe feature is enabled
4	wheel	0	touch wheel for rotation is disabled
		1	touch wheel for rotation is enabled
3	SK	0	soft key and Touch Button functionality is disabled
		1	soft key and Touch Button functionality is enabled
2	ALS	0	ambient light sensor is off
		1	ambient light sensor is on
1	LED	0	LED off
		1	LED on
0	display	0	Display off
		1	Display on

Operation mode	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Start-up or inactive mode	0	0	0	0	0	0	0	0
Sleep mode	0	0	x	x	1	0	0	0
Active mode	0	0	x	x	x	x	x	1
screen saver mode	0	0	x	1	1	x	1	0

6.3 Setting the brightness of the display

DISPLAY_CFG

Name	DISPLAY_CFG							
Addr	0x12							
Bit	7	6	5	4	3	2	1	0
r	reserved				brightness			
w								
Reset	0x0F							

Bit	Symbol	Value	Description
7 to 4	reserved	0x0	Do not use
3 to 0	brightness	0x0 to 0xF	Brightness of the display, where 0xF correspond to the maximum brightness of 100% and 0x0 corresponds to 6.3% of the maximum brightness of the display

6.4 Configuration of the Home Button LED

The soft key at 6 o'clock is marked with a home symbol which can be backlighted by an RGB LED. This Home Button LED offers a big number of illumination colors which can be activated in three different modes: constant illumination, blinking, or pumping mode. The registers LED_CFG, LED_COLOR_R, LED_COLOR_G and LED_COLOR_B can be used to set the LED into the desired mode.

LED_CFG

Name	LED_CFG							
Addr	0x13							
Bit	7	6	5	4	3	2	1	0
r	reserved		pattern		brightness			
w								
Reset	0x0F							

Bit	Symbol	Value	Description
7 to 6	reserved	0	Do not use
5 to 4	pattern	00	constantly LED illumination
		01	LED is blinking
		10	LED is pumping
		11	Reserved, do not use
3 to 0	brightness	0x0 to 0xF	Brightness of the LED, where 0xF correspond to the maximum brightness of 100% and 0x0 corresponds to 6.3% of the maximum brightness of the LED

LED_COLOR_R

Name	LED_COLOR_R							
Addr	0x14							
Bit	7	6	5	4	3	2	1	0
r	intensity red							
w								
Reset	0x00							

Bit	Symbol	Value	Description
7 to 0	intensity red	0x00 to 0xFF	Intensity of the red color for the LED

LED_COLOR_G

Name	LED_COLOR_G							
Addr	0x15							
Bit	7	6	5	4	3	2	1	0
r	intensity green							
w								
Reset	0x00							

Bit	Symbol	Value	Description
7 to 0	intensity green	0x00 to 0xFF	Intensity of the green color for the LED

LED_COLOR_B

Name	LED_COLOR_B							
Addr	0x16							
Bit	7	6	5	4	3	2	1	0
r	intensity blue							
w								
Reset	0x00							

Bit	Symbol	Value	Description
7 to 0	intensity blue	0x00 to 0xFF	Intensity of the blue color for the LED

6.5 Configuration of the ambient light sensor ALS_STATUS

The CDS1 has an integrated ambient light sensor which is located at the 12 o'clock position under the blank part of the glass. The ambient light sensor can be used to synchronize the brightness of the display and of the Home Button LED. It is recommended to use pictures with dark background colors and to reduce the brightness of the display when the CDS1 is not in use for longer time. This can help to reduce the aging effect of the OLED display.

The current value of the ambient light sensor can be read from the register ALS_STATUS.

Name	ALS_STATUS							
Addr	0x07							
Bit	7	6	5	4	3	2	1	0
r	ambient light value (LSByte)							
w	-							
Reset	0x00							
Addr	0x08							
Bit	7	6	5	4	3	2	1	0
r	ambient light value (MSByte)							
w	-							
Reset	0x00							

Bit	Symbol	Value	Description
7 to 0	ambient light value (LSByte)	0x00 to 0xFF	Contains the value of the ambient light sensor 0xYYZZ, LSByte = ZZ
7 to 0	ambient light value (MSByte)	0x00 to 0xFF	Contains the value of the ambient light sensor 0xYYZZ, MSByte = YY

6.6 Configuration of pictures and animated gif videos

The CDS1 has an internal mass storage which allows to store up to 200 pictures on the device. The actual number of pictures that can be stored on the CDS1 depends on the file sizes of the pictures and animated gif files and therefore can be less than 200 pictures.

All pictures have to follow the respective naming convention rules as explained below. Pictures that do not follow these rules will be ignored by the CDS1.

The upload of pictures and animated gif videos will be done via the USB interface.

The CDS1 offers four main operating modes which are

1. no picture (black display)
2. display a freeze image
3. scroll rotation pictures
4. display animated gif videos

The register SWITCH_MODE is used to enable or disable either of these operating modes and to set generally options related to them.

SWITCH_MODE

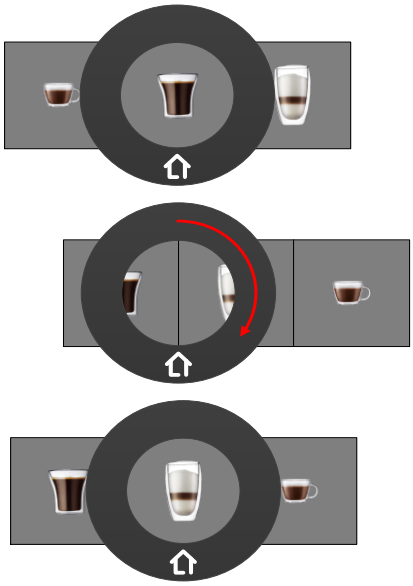
Name	SWITCH_MODE							
Addr	0x20							
Bit	7	6	5	4	3	2	1	0
r	screen saver	mode parameter			reserved	config mode	mode	
w								
Reset	0x00							

Bit	Symbol	Value	Description
7	screen saver	0	operation mode
		1	Screen saver mode: the display becomes black
6 to 4	mode parameter		When mode = 00: in this case the mode parameter does not have any function
			When mode = 01:
		000	no partial picture
		001	1 partial picture
		010	2 partial pictures
		011	3 partial pictures
		100	4 partial pictures
			When mode = 10:
		000	Rotation or slide left and right
		001	Rotation or slide up and down
		010	snap scroll
			When mode = 11:
		000	endless video
		001	standard video, stops after first run
3	reserved	0	Do not use
2	configmode		when configuration is set to "USB serial"
		0	normal operating mode according to the configuration
		1	configuration mode: access to the USB mass storage is activated. **Attention: Reset is only possible via a power cycle!**
1 to 0	mode		display mode setting Note: in case an image is not existing or wrong frame rate for an animated gif, an error message is shown on the display.
		00	black image
		01	image is frozen
		10	rotation or slide mode
		11	video mode

There are 3 different scroll modes possible:

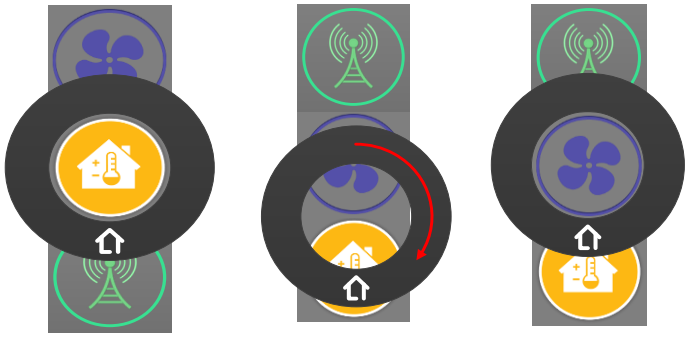
1. Slide scroll horizontal

The rotation pictures slide continuously through the display area in horizontal direction. During the transition from one picture to the next picture, parts of both pictures are visible.



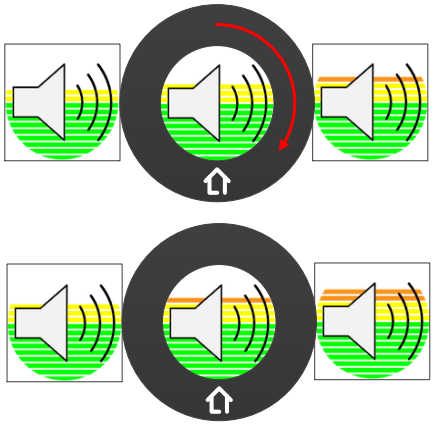
2. Slide scroll vertical

The rotation pictures slide continuously through the display area in vertical direction. During the transition from one picture to the next picture, parts of both pictures are visible.



3. Snap scroll

The rotation change from one picture to the next picture is a sudden change. There is always only one full picture visible on the display.



The scroll direction is only related to the change direction of the pictures while the swipe direction can be either horizontally or vertically.

6.6.1 Pictures

The CDS1 accepts only pictures in *.png format which fulfill the following rules:
- The file format is *.png
- 128 x 128 pixels, only partial pictures are allowed to be smaller
- Maximum file size of 20 kByte; File compressing can be done i.e. with www.tinypng.com

Pictures which do not fulfill these conditions are ignored by the CDS1 and cause an error message on the display.

The png pictures may show transparency to the overlapping of pictures. Pixels with a part transparency are processed by the CDS1 as 100% transparent.

Background pictures

Naming conventions for background pictures:

B	00	picture number	optional individual description

B	picture category is „background picture“
00	has to be always „00“
picture number	00 to FF (hexadecimal numbering), where 00 is reserved as black background; background pictures with picture number 00 are ignored by the CDS1
individual description	free choice text, can also be empty

Example:
B0001_my_background_picture

Each background picture must consist of 128 x 128 pixel

BACKGROUND

Name	BACKGROUND							
Addr	0x23							
Bit	7	6	5	4	3	2	1	0
r	picture number (LSByte)							
w								
Reset	0x00							
Addr	0x24							
Bit	7	6	5	4	3	2	1	0
r	reserved (MSByte)							
w								
Reset	0x00							

Bit	Symbol	Value	Description
7 to 0	picture number (LSByte)	0x00	background is black
		0x01 to 0xFF	background picture number
7 to 0	reserved (MSByte)	0x0	Do not use

Rotation pictures

Naming conventions for rotation pictures:

R	rotation group number	picture number	optional individual description

R	picture category is „rotation picture“
rotation group number	00 to FF (hexadecimal numbering)
picture number	00 to FF (hexadecimal numbering)
individual description	free choice text, can also be empty

Examples:
R0A01_my_rotation_start_picture
R0A02_my_next_rotation_picture

The CDS1 starts the rotation with the picture that is specified in register ROTATION (address 0x21). The rotation sequence of rotation pictures is defined by increasing numbering of the pictures and must not have any gaps in the picture numbering within a rotation group. If there is a picture number missing in the sequence, the rotation sequence excludes all pictures that have higher picture numbers than the missing picture within the rotation group.

A rotation picture can be a combination from a background picture with a foreground picture. While the background picture must be a 128 x 128 pixel picture, the foreground picture can be a partial picture. In case the foreground picture has the size 128 x 128 pixels, no background picture is needed.

The following registers are used to define the conditions for the rotation pictures.

ROTATION

Name	ROTATION							
Addr	0x21							
Bit	7	6	5	4	3	2	1	0
r w	start picture number (LSByte)							
Reset	0x00							
Addr	0x22							
Bit	7	6	5	4	3	2	1	0
r w	rotation group (MSByte)							
Reset	0x00							

Bit	Symbol	Value	Description
7 to 0	start picture number (LSByte)	0x0	Number of the picture to be shown in rotation mode
7 to 0	rotation group (MSByte)	0x0	rotation group to be shown

TOUCH_CFG

Name	TOUCH_CFG							
Addr	0x17							
Bit	7	6	5	4	3	2	1	0
r w	reserved				snap scroll sensitivity		rotation speed	
Reset	0x0B							

Bit	Symbol	Value	Description
7 to 4	reserved	0x0	Do not use
3 to 2	snap scroll sensitivity		Definition of the delicacy for the picture change, when the mode in the rotation menu is snap scroll
		00	15° is required for a snap scroll from one picture to the next one
		01	30° is required for a snap scroll from one picture to the next one
		10	45° is required for a snap scroll from one picture to the next one
		11	60° is required for a snap scroll from one picture to the next one
1 to 0	rotation speed		Definition of the rotation speed, when the mode in the rotation menu is either horizontal or vertical
		00	use 45° for the rotation of one full picture
		01	use 60° for the rotation of one full picture
		10	use 75° for the rotation of one full picture
		11	use 90° for the rotation of one full picture

Get the status of the pictures
MENU_STATUS

This register indicates the active picture number and the rotation group within the rotation menu.

Name	MENU_STATUS							
Addr	0x05							
Bit	7	6	5	4	3	2	1	0
r	current picture (LSByte)							
w	-							
Reset	0x00							
Addr	current rotation group (MSByte)							
Bit	7	6	5	4	3	2	1	0
r	current rotation group (MSByte)							
w	-							
Reset	0x00							

Bit	Symbol	Value	Description
7 to 0	current picture (LSByte)		Contains the number of the current image after the interrupt (LSByte = ZZ)
7 to 0	current rotation group (MSByte)		Contains the number of the rotation group after the interrupt (MSByte = YY) RYYZZ

Partial pictures

A background picture can be combined with up to 4 partial pictures. These partial pictures can be of smaller size than 128 x 128 pixels. Each of them can be located anywhere on the display.

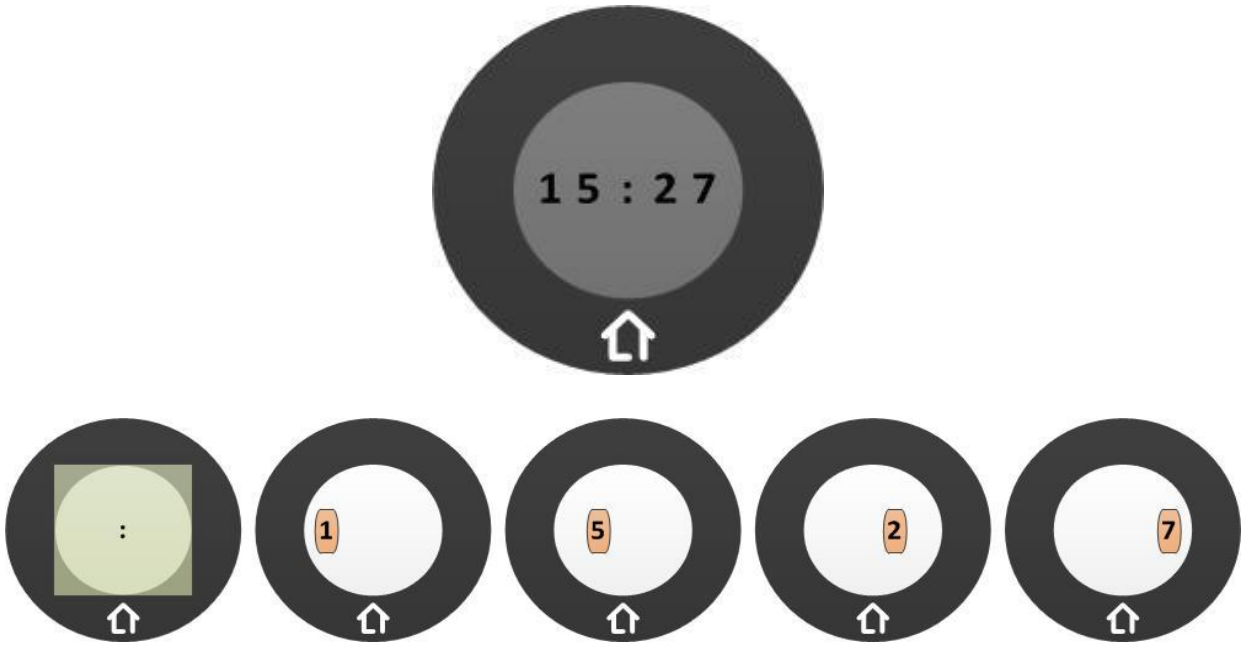
Naming conventions for partial pictures:

P	00	picture number	optional individual description

P	picture category is „partial picture“
00	has to be always „00“
picture number	00 to FF (hexadecimal numbering)
individual description	free choice text, can also be empty

Example:
P0001_my_partial_picture

Example of a clock display consisting of a background picture and four partial pictures:



background picture | partial picture 1 | partial picture 2 | partial picture 3 | partial picture 4

The following registers are used to define the conditions for the partial pictures und their location on the display.

PART1

Name	PART1							
Addr	0x25							
Bit	7	6	5	4	3	2	1	0
r w	number of the first partial image (LSByte)							
Reset	0x00							
Addr	0x26							
Bit	7	6	5	4	3	2	1	0
r w	reserved (MSByte)							
Reset	0x00							

Name	PART1 POSITION							
Addr	0x27							
Bit	7	6	5	4	3	2	1	0
r w	reserved	X position of partial image number 1 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							
Addr	0x28							
Bit	7	6	5	4	3	2	1	0
r w	reserved	Y position of partial image number 1 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							

PART2

Name	PART2							
Addr	0x29							
Bit	7	6	5	4	3	2	1	0
r w	number of the second partial image (LSByte)							
Reset	0x00							
Addr	0x2A							
Bit	7	6	5	4	3	2	1	0
r w	reserved (MSByte)							
Reset	0x00							

Name	PART2 POSITION							
Addr	0x2B							
Bit	7	6	5	4	3	2	1	0
r w	reserved	X position of partial image number 2 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							
Addr	0x2C							
Bit	7	6	5	4	3	2	1	0
r w	reserved	Y position of partial image number 2 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							

PART3

Name	PART3							
Addr	0x2D							
Bit	7	6	5	4	3	2	1	0
r w	number of the third partial image (LSByte)							
Reset	0x00							
Addr	0x2E							
Bit	7	6	5	4	3	2	1	0
r w	reserved (MSByte)							
Reset	0x00							

Name	PART3 POSITION							
Addr	0x2F							
Bit	7	6	5	4	3	2	1	0
r w	reserved	X position of partial image number 3 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							
Addr	0x30							
Bit	7	6	5	4	3	2	1	0
r w	reserved	Y position of partial image number 3 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							

PART4

Name	PART4							
Addr	0x31							
Bit	7	6	5	4	3	2	1	0
r w	number of the fourth partial image (LSByte)							
Reset	0x00							
Addr	0x32							
Bit	7	6	5	4	3	2	1	0
r w	reserved (MSByte)							
Reset	0x00							

Name	PART4 POSITION							
Addr	0x33							
Bit	7	6	5	4	3	2	1	0
r w	reserved	X position of partial image number 4 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							
Addr	0x34							
Bit	7	6	5	4	3	2	1	0
r w	reserved	Y position of partial image number 4 (0 - 127) starting point is in the upper left edge of the display						
Reset	0x00							

6.6.2 Animated gif videos

The CDS1 supports animated gif videos. The animated gif videos are built from a sequence of pictures which have each a size of 128 x 128 pixels. The following restrictions have to be fulfilled:

- The file format is *.gif
- 128 x 128 pixels
- Minimum frame rate is 60 ms
- Maximum file size is 128 kByte; The files can be compressed by i.e. ezgif.com/optimize

Animated gif videos which do not fulfill these conditions are ignored by the CDS1 and cause an error message on the display.

A video can be started as a one-time-run or in an endless loop. Which option is chosen can be defined in the register SWITCH_MODE.

VIDEO

Name	VIDEO							
Addr	0x35							
Bit	7	6	5	4	3	2	1	0
r w	video number (LSByte)							
Reset	0x00							
Addr	0x36							
Bit	7	6	5	4	3	2	1	0
r w	reserved (MSByte)							
Reset	0x00							

6.7 Touch Button and Soft Keys

Status of the Touch Button BUTTON_STATUS

Name	BUTTON_STATUS							
Addr	0x03							
Bit	7	6	5	4	3	2	1	0
r	reserved						DBL	DBS
w	-						-	-
Reset	0x00							

Bit	Symbol	Value	Description
7 to 2	reserved	000000	Do not use
1	DBL	0	No key stroke on the touch button
		1	Long key stroke on the touch button
0	DBS	0	No key stroke on the touch button
		1	Short key stroke on the touch button

Status of the Soft Keys SOFTKEY_STATUS

Name	SOFTKEY_STATUS							
Addr	0x04							
Bit	7	6	5	4	3	2	1	0
r	RIGHT_L	TOP_L	LEFT_L	BOTTOM_L	RIGHT_S	TOP_S	LEFT_S	BOTTOM_S
w	-	-	-	-	-	-	-	-
Reset	0x00							

Bit	Symbol	Value	Description
7	RIGHT_L	0	No key stroke on the 3 o'clock soft key
		1	Long key stroke on the 3 o'clock soft key
6	TOP_L	0	No key stroke on the 12 o'clock soft key
		1	Long key stroke on the 12 o'clock soft key
5	LEFT_L	0	No key stroke on the 9 o'clock soft key
		1	Long key stroke on the 9 o'clock soft key
4	BOTTOM_L	0	No key stroke on the 6 o'clock soft key
		1	Long key stroke on the 6 o'clock soft key
3	RIGHT_S	0	No key stroke on the 3 o'clock soft key
		1	Short key stroke on the 3 o'clock soft key
2	TOP_S	0	No key stroke on the 12 o'clock soft key
		1	Short key stroke on the 12 o'clock soft key
1	LEFT_S	0	No keystroke on the 9 o'clock soft key
		1	Short keystroke on the 9 o'clock soft key
0	BOTTOM_S	0	No keystroke on the 6 o'clock soft key
		1	Short keystroke on the 6 o'clock soft key

TOUCH_STATUS

The touch wheel area is divided into 256 segments. This results in an angular resolution of approximately 1.4 degrees per segment. The segments on the touch wheel are numbered counter-clockwise from 0 to 255. The register TOUCH_STATUS contains the number of the last touched segment on the touch wheel.

Name	TOUCH_STATUS							
Addr	0xF6							
Bit	7	6	5	4	3	2	1	0
r	last pressed position on the touch wheel							
w	-							
Reset	0x00							

6.8 Special registers

Readout the status of the interrupt flags

INTERRUPT_FLAGS

Name	INTERRUPT_FLAGS							
Addr	0x02							
Bit	7	6	5	4	3	2	1	0
r	Reset/ Error	CRC Error	reserved	VID	Scroll	SK	TB	sum
w	-	-	-	-	-	-	-	-
Reset	0x81							

Bit	Symbol	Value	Description
7	Reset/Error	0	No reset, no error
		1	A reset was applied on the LCD Switch. The reason for the reset is stored in the register RESET_REASON
6	CRC Error	0	No CRC error
		1	Wrong CRC checksum of a machine instruction
5	reserved	0	Do not use
4	VID	0	no video active
		1	A video was aborted or has reached ist end
3	Scroll	0	No rotation or swiping activity was detected
		1	A rotational or swiping motion was carried out
2	SK	0	No soft-key was activated
		1	A keystroke on a soft-key was applied. The Information about which Soft Key was pressed and about how long the soft-key was pressed is stored in the register SOFTKEY_STATUS.
1	TB	0	The touch button was not activated
		1	A keystroke on the Touch-Button happened. The Information on the duration of the keystroke is stored in the register BUTTON_STATUS
0	sum	0	No interrupt has happened
		1	At least one flag was set

The interrupt flags for the rotation, end of video and reset are reset after the register INTERRUPT_FLAGS is read.

Get the Interrupt status

IRQ_CFG

Name	IRQ_CFG							
Addr	0x11							
Bit	7	6	5	4	3	2	1	0
r w	reserved	Reset / Error	CRC	reserved	VID	Scroll	SK	TB
Reset	0x6F							

Bit	Symbol	Value	Description
7	reserved	0	Do not use
6	Reset / Error	0	No reset or error occurred.
		1	A reset or an error event occurred. Check the register ERROR to get further information if this flag was set due to a reset or due to an error and get further information on the error reason.
5	CRC	0	CRC Error is disabled
		1	CRC Error is enabled
4	reserved	0	Do not use
3	VID	0	Video Interrupt is disabled
		1	Video Interrupt is enabled
2	Scroll	0	Interrupt from rotation or slide activities is disabled
		1	Interrupt from rotation or slide activities is enabled
1	SK	0	Interrupt from the soft keys is disabled
		1	Interrupt from the soft keys is enabled
0	TB	0	Interrupt from the touch button is disabled
		1	Interrupt from the touch button is enabled

ERROR

The register ERROR provides details about the reason for an error. This register is read only.

Name	ERROR							
Addr	0xF7							
Bit	7	6	5	4	3	2	1	0
r	Error Code							
w	-							
Reset	0x00							

Error Code	Category	Description	Module	Operating Mode	Actions Reset
0x00	Reset	Reset happened / no error	-	-	No
0x01	General	Unknown error	-	-	No
*	*	*	Machine Interface	Inactive	Yes
*	*	*	Machine Interface	Sleep / Screen saver	No
*	*	*	Machine Interface	Active	No
*	*	PNG / GIF decompression error due to not being able to allocate enough memory	View Handler	Active	Yes
0x02	RTOS	Unknown error with operating system	-	-	No
0x03	RTOS	Mutex error	-	-	No
0x04	RTOS	Signal error	-	-	No
0x05	RTOS	Signal timeout	-	-	No
0x06	General	Out of range error	-	-	No
0x07	General	Out of upper range error	-	-	No
0x08	General	Our of lower range error	-	-	No
0x09	General	Write protection error	-	-	No
0x0A	General	Read protection error	-	-	No
0x0B	General	Initialization error	-	-	No
*	*	*	Display	Active / Screen saver	No
*	*	*	Touch Controller	Active	No
*	*	*	Touch Controller	Screen saver	No
*	*	*	Touch Controller	Inactive / Sleep	No
0x0C	General	Invalid parameter error	-	-	No
0x0D	General	Parameter not found error	-	-	No
0x0E	General	File error	-	-	No
*	*	File of image not found	Image Loader	Active	Yes
0x0F	General	Open file error	-	-	No
*	*	*	Image Loader	Active	No
0x10	General	Close file error	-	-	No
0x11	General	Unhandled state error	-	-	No
0x12	General	Unhandled event error	-	-	No
0x13	General	Buffer overflow error	-	-	No
0x14	General	Buffer underrun error	-	-	No

RESET_REASON

This register contains detailed information about the reason for a reset.

Name	RESET_REASON							
Addr	0xF8							
Bit	7	6	5	4	3	2	1	0
r	LPWR RSTF	WWDG RSTF	IWDG RSTF	SFT RSTF	POR RSTF	PIN RSTF	BORRS TF	-
w	-							
Reset	0x00							

Bit	Symbol	Value	Description
7	LPWR RSTF		Low-power management reset
6	WWDG RSTF		Window watchdog reset flag
5	IWDG RSTF		Independent watchdog reset flag
4	SFT RSTF		Software reset flag
3	POR RSTF		POR / PDR reset flag
2	PIN RSTF		PIN reset flag
1	BORRS TF		BOR reset flag

DEVICE_ACTIONS

This register will be set in case the CDS needs to be restarted.

Name	DEVICE_ACTIONS							
Addr	0xFA							
Bit	7	6	5	4	3	2	1	0
r	reserved							RESET
w								
Reset	0x00							

Bit	Symbol	Value	Description
0	RESET	0	no reset required
		1	do a restart of the CDS1

SERIAL_NR

This register contains the serial number of your CDS1 device. It is a code which is unique to your devices and allows to trace back production details.

Name	SERIAL_NR							
Addr	0xFC							
Bit	7	6	5	4	3	2	1	0
r	Serial Number Byte 0							
w	-							
Reset	-							
Addr	0xFD							
Bit	7	6	5	4	3	2	1	0
r	Serial Number Byte 1							
w	-							
Reset	-							
Addr	0xFE							
Bit	7	6	5	4	3	2	1	0
r	Serial Number Byte 2							
w	-							
Reset	-							
Addr	0xFF							
Bit	7	6	5	4	3	2	1	0
r	Serial Number Byte 3							
w	-							
Reset	-							

7 Cleaning instructions

To preserve the attractive look of your coated glass, you must clean it on a regular basis just like any other type of glazing. How often it needs to be cleaned depends on the location and how dirty it is.

Appropriate cleaning measures

Normal dirt can be removed using the usual wet cleaning processes. Here, you will need water, a sponge, a cloth, a wiper or chamois leather. When doing so, only neutral wetting agents that do not contain any abrasive additives should be added to the water.
Stubborn residues (e.g. color or adhesive residues, tar splashes, etc.) should be removed using suitable solvents (see table). Here, you should make sure that the sealing materials of the glazing are not damaged.

Clean tools such as:
- Cloths
- Chamois leather
- Sponges
- Wipers
- Plenty of water

With normal soiling:
- Water (demineralized or city water)
- Cleaners that contain surfactants such as Alklar (ECOLAB), Pril (HENKEL)
- Cleaners that contain ammonia such as AJAX glass cleaner (Colgate Palmolive)
- Mild alkaline cleaning agents such as deconex FPD 111 (BORER)

Solvents:
- Acetone (not to be used with NARIMA®, DARO!)
- Ethanol
- Benzine or cleaning benzine (e.g. Centralin® household benzine)
- Graffiti cleaners that contain solvents (e.g. GRAFFINET® cleaner)

Inappropriate cleaning measures

All strong alkaline detergent solutions and acids are unsuitable, especially hydrofluoric acid or cleaning agents that contain fluoride. They attack the anti-reflective glass surfaces and cause irreparable chemical burns. Rough cleaning agents (e.g. abrasives, steel wool, blades, etc.) should not be used either.

Do not use the following for cleaning:
- Strongly alkaline detergent solutions
- Acids, e.g. hydrofluoric acid
- Detergents that contain fluoride

Rough cleaning agents (strongly abrasive cleaners) such as:
- Abrasive cleaners and cleaning agents that contain abrasive milk
- Steel wool
- Scrubbing sponges
- Blades
- Fabrics with in-woven metal wires
- Pumice powder

SCHURTER AG
Werkhofstrasse 8 - 12
Postfach
CH - 6002 Luzern
schurter.com