



The Third Evolution™



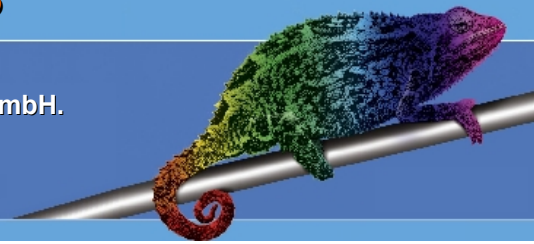
Programmable,
RGB-backlit
LCD Keyswitches

APPLICATION NOTE

Version 2.1

READ COMMANDS

© 2004-2006 copyright [E³] Engstler Elektronik Entwicklung GmbH.
All rights reserved.



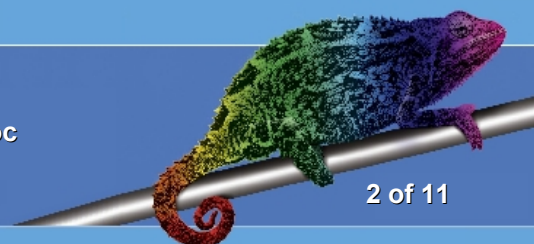
READ Commands

Application Note



Table of Content

Application Note.....	1
READ Commands.....	1
Table of Content.....	2
General Description.....	3
Command Examples.....	4
Special Considerations.....	4
Sample Transmission.....	5
Source Sample for Receiving Serial Data from Key.....	7
Partial Schematic of the DemoBoard.....	10
Change History.....	11
Contact Information.....	12



READ Commands

Application Note

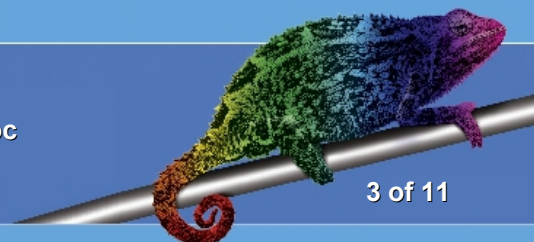


General Description

The SA switch family includes the SA3216, SA3624 and SA6432 LCD Keyswitches with command driven serial interfaces. They integrate a graphical liquid crystal display with **RGB** backlighting in a keyswitch. The SA keys are controlled via a serial interface to the integrated *Advanced Technology*[™] electronics, which control the interface, display and backlighting. SA keys self-initialise without external setup commands. Data is only transmitted when a change is made to the display or background colours. Only six contact terminals are required to provide power, clock and data lines as well as switch contacts. The contact pins of the internal switch are isolated from the internal electronics. This documents describes the “**READ COMMANDS (ID & Serial Number)**”. The purpose of this enhancement is it to enable customer systems to automatically detect which version is in use since identical hardware set-ups can be used for SA3216, SA3624 and SA6432 switches.

Command <i>(binary representation)</i>	Command Name / Description	Comments
01000100 (0x44)	Read Keyswitch ID* THIS COMMAND FORCES THE KEYSWITCH TO ANSWER ON THE CLOCK AND DATA LINE. (The clock is generated by the SA switch; see Application Notes at www.e3-keys.com)	The answer is consisting of ASCII characters representing the Keyswitch ID and is terminated with CR (0x0D): SA3216 SA3624 SA6432 SB6432 (not supported in <i>Legacy Mode</i> [™])
01000101 (0x45)	Read Serial Number * THIS COMMAND FORCES THE KEYSWITCH TO ANSWER ON THE CLOCK AND DATA LINE. (The clock is generated by the SA switch; see Application Notes at www.e3-keys.com)	The answer is consisting of 4 Bytes which give the serial Number in the following format and is terminated with CR (0x0D): SNYYWW##### Year (04-99) Week (01-52) Number (00000 .. 99999) (not supported in <i>Legacy Mode</i> [™])

* To take advantage of these advanced functions your hardware must ensure that the SA keys are actively driving the serial data lines in *Advanced Technology*[™] mode.





Command Examples

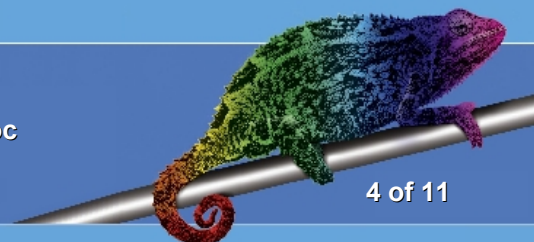
Read Keyswitch ID: 01001000 (0x44)		
<i>Binary</i>	<i>HEX</i>	<i>Comments</i>
01001000	0x44	Read Keyswitch ID
The Key will answer by generating its own clock and data signals The following examples show the answers for the different SAxxxx types: 0x53 0x41 0x33 0x32 0x31 0x36 0x0D = SA3216 CR 0x53 0x41 0x33 0x36 0x32 0x34 0x0D = SA3624 CR 0x53 0x41 0x36 0x34 0x33 0x32 0x0D = SA6432 CR		

Read Serial Number: 01001001 (0x49)		
<i>Binary</i>	<i>HEX</i>	<i>Comments</i>
01001001	0x45	Read Serial Number (SNYYWW#####)
The Key will answer by generating its own clock and data signals The following example shows the format of the answer: 0x53 0x4E 0x30 0x34 0x30 0x33 0x30 0x35 0x30 0x39 0x33 0x0D = SN040305093CR		

Special Considerations

Since the keyswitches will respond on their own to these 2 commands special considerations have to be taken when designing the hard- and software.

In order to minimize the hardware and software restraints the keyswitches will respond regardless of the prior communication speed with a relatively low speed. The issued clock frequency will be roughly about 10kHz ensuring that even slow microprocessors should be able to read back the data.



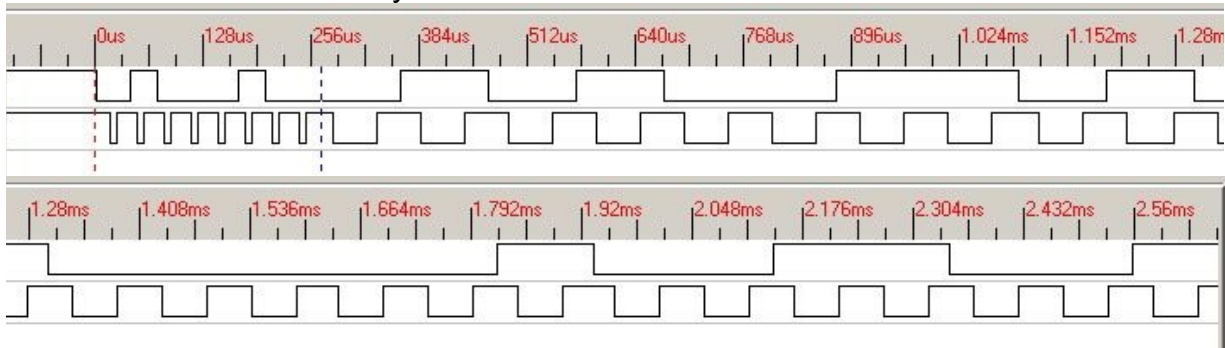
READ Commands

Application Note



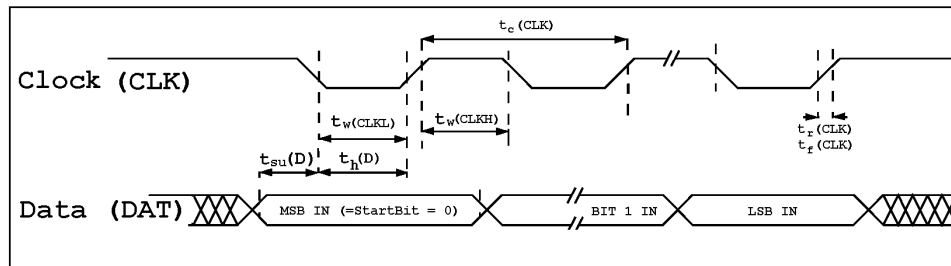
Sample Transmission

The following diagram shows the clock line taken by logic analyzer from an actual data transmission. Please note that the transmission is only partially shown and starts with the issued command “Read Keyswitch ID”:

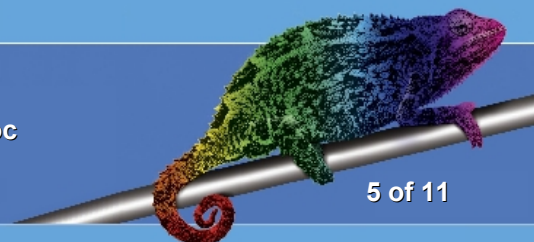


For detailed timing descriptions for signal phase and general timing please review the corresponding section in the datasheet.

TIMING DIAGRAM FOR SIGNALS GENERATED BY SAxxxx KEYS



Symbol	Parameter	Min	Max	Unit
$t_c(\text{CLK})$	Key generated CLK	9	11	kHz
$t_w(\text{CLKH})$	Clock high time	40	60	μs
$t_w(\text{CLKL})$	Clock low time	40	60	μs
$t_{su}(\text{D})$	Data input setup time	140		ns
$t_h(\text{D})$	Data input hold time	100		ns
$t_r(\text{CLK})$	Clock rise time		25	ns
$t_f(\text{CLK})$	Clock fall time		25	ns





Source Sample for Receiving Serial Data from Key

The following source is an excerpt of the DemoBoard firmware. The DemoBoard is controlled by a PIC16F627 controller. The relevant part of the schematic is shown following the source listing.

```
;
;*****
;          SAREad
;*****
;
SAREad:          ; Read Data = W from SA Key
                 ;          = and SABuf
;   SA_CLOCK_Stopp          ; Stop permanent clock on LCD Keys if active

                 ; functions
movlw  b'11101111'    ; I: RA0 Key1
                 ; I: RA1 Key2
                 ; I: RA2 Key1Data
                 ; I: RA3 Key2Data
                 ; O: RA4 PowerON
                 ; I: RA5 VPP
                 ; I: RA6 CLK
                 ; I: RA7 CLK

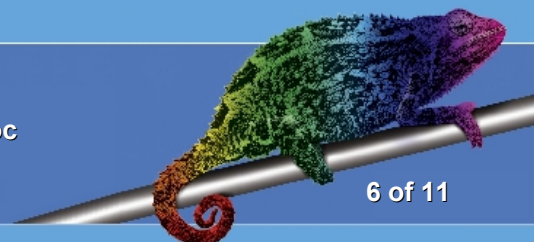
movwf  PortAMask     ; Save Default TRISA Setting
banksel  TRISA
movwf  TRISA

                 ;
movlw  b'11001011'    ; I: RB0 PowerFail
                 ; I: RB1 RxD
                 ; O: RB2 TxD
                 ; I: RB3 LCDClock
                 ; O: RB4 EEClock
                 ; O: RB5 EEDATA
                 ; I: RB6 PGClock
                 ; I: RB7 PGData

movwf  TRISB
banksel  PortBMask
movwf  PortBMask     ; Save Default TRISB Setting
banksel  PORTB

rdloop
movlw  .8
movwf  SABitCount    ; set the #bits to 8
clrf  Time           ; Reset Time to check for ReadError

bitin
btfsc  Time,2        ; wait for clock high
                 ; if timeout back with error
```



READ Commands

Application Note



```
        goto rderr
        btfss PORTB,3      ; wait for clock high
        goto bitin
wclklow
        btfsc Time,2      ; if timeout back with error
        goto rderr
        btfsc PORTB,3      ; wait for clock low to shift in data
        goto wclklow      ; if not wait again for clock low

sard
        clrfs Time        ; Reset Timeout

        bcf STATUS,C      ; clear Carry

        btfss KeyMask, Key1 ; if Key1 selected
        goto bitink2
        btfsc PORTA, Key1  ; set carry equal to Key1
        bsf STATUS,C
        goto shiftin
bitink2
        btfss KeyMask, Key2 ; if Key2 selected
        goto shiftin
        btfsc PORTA, Key2  ; set carry equal to Key1
        bsf STATUS,C
shiftin
        rlf SABuf,f       ; shift in bit

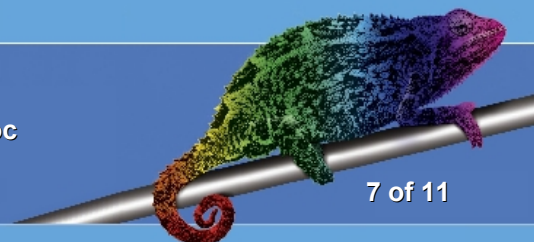
sard2
        decfsz SABitCount, F ; 8 bits done?
        goto bitin        ; no - nxt bit

        movf SABuf,W      ;
        goto SAREadx

rderr
        clrfs SABuf
        goto SAREadx2

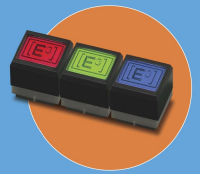
SAREadx
        call TX
        movf SABuf,w      ;
        sublw 0x0d
        btfss STATUS,Z
        goto rdloop
SAREadx2
        movlw 0x0a        ; new line
        call TX

        movlw b'11100011' ; functions
                           ; I: RA0 Key1
                           ; I: RA1 Key2
                           ; O: RA2 Key1Data
                           ; O: RA3 Key2Data
                           ; O: RA4 PowerON
```



READ Commands

Application Note



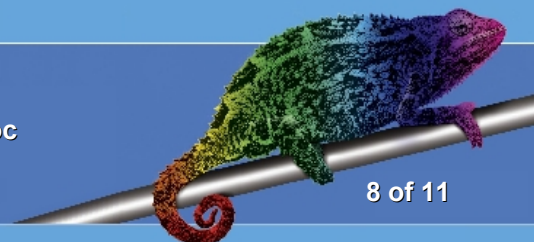
```

; I: RA5 VPP
; I: RA6 CLK
; I: RA7 CLK
movwf PortAMask      ; Save Default TRISA Setting
banksel TRISA
movwf TRISA

;
movlw b'11000011'   ; I: RB0 PowerFail
; I: RB1 RxD
; O: RB2 TxD
; O: RB3 LCDClock
; O: RB4 EEClock
; O: RB5 EEDATA
; I: RB6 PGClock
; I: RB7 PGData

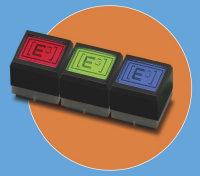
movwf TRISB
banksel PortBMask
movwf PortBMask     ; Save Default TRISB Setting

return
```

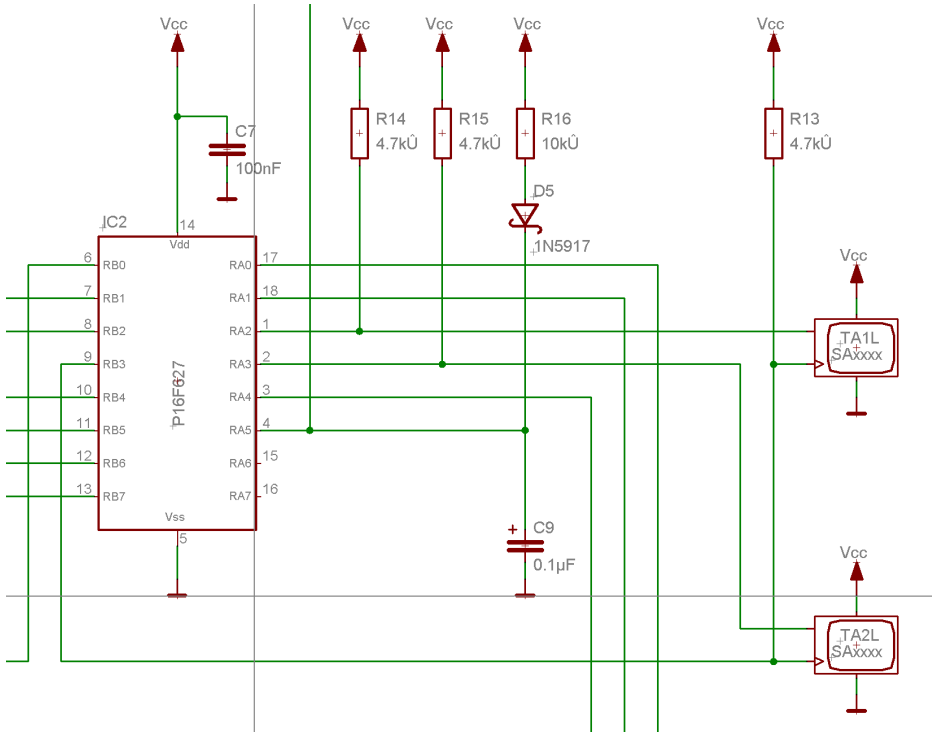


READ Commands

Application Note



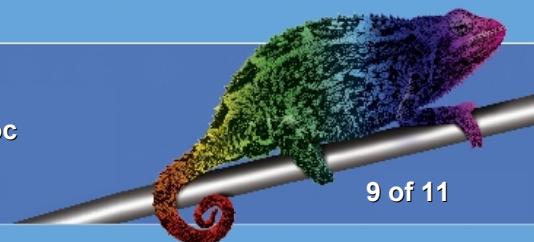
Partial Schematic of the DemoBoard

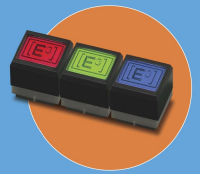


The SAxxxx keys are directly connected to the PIC16F627 controller. In this schematic we just see the clock and data lines of the SAxxxx switches. The contact matrix is not shown since it is not relevant for the communication.

The Clock signal is generated on PortPin RB3 of the PIC16F627 and is common for both SAxxxx keys.

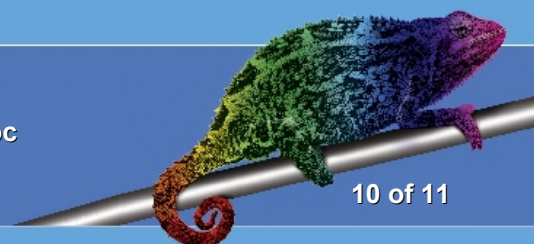
Data is sent and received through RA2 and RA3 respectively of the PIC16F627 microcontroller.





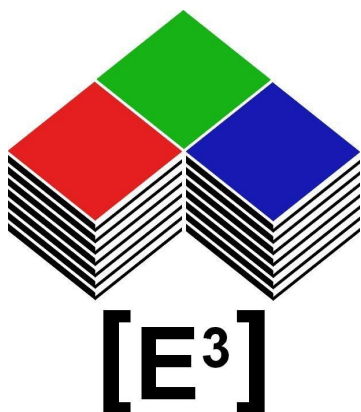
Change History

<i>Version</i>	<i>Date</i>	<i>Comments</i>
0.1	03/08/04	Initial draft document derived from Technical Datasheet 1.1
1.0	06/07/04	Release
2.0	11/01/05	Updated document and layout
2.1	03/14/06	Updated layout
2.2	02/15/11	Command Ids corrected





Contact Information



[E³]
Engstler
Elektronik
Entwicklung
GmbH

Industriering 7
63868 Grosswallstadt
Germany

Phone: +49 (0) 6022 262570
Fax: +49 (0) 6022 262571
E-Mail: info@e3-keys.com

