



PIC-Quellcode

Anwendungshinweis

Inhaltsverzeichnis

PIC-Quellcode	3
Hinweise	8
Copyright Hinweis.....	8
Technische Hinweise.....	8
Gewährleistungsausschluss.....	8
Hinweis zur deutschen Version.....	9
Änderungshistorie	10

PIC-QUELLCODE

Der folgende Assembler-Quelltext stammt aus unserer Firmware, die in unserem DS00000 DemoBoard Prototyping Kit verwendet wird. Das DemoBoard wird von einem PIC16F627/8 betrieben.

```
*****
;
;           Sample Code, derived from DemoBoard           *
*****
;
;   Filename:      Sample source PIC.asm                 *
;   Date:         13.02.2004                             *
;   File Version:  First Revision 26.01.2004             *
;   Author:       Reinhard Engstler                     *
;   Company:      [E3] Engstler Elektronik Entwicklung GmbH *
*****
;
;   Files required: p16f628.inc                          *
*****
;
;   Notes:  actual code snippets not tested              *
;           just cut and paste from working demoboard   *
*****

        list      p=16f628      ; list directive to define processor
        #include  p16f628.inc    ; processor specific variable definitions

__CONFIG _CP_OFF & _WDT_OFF & _BODEN_ON & _PWRTE_ON & _INTRC_OSC_NOCLKOUT
& _MCLRE_OFF & _LVP_OFF

; '__CONFIG' directive is used to embed configuration data within .asm
file.
; The labels following the directive are located in the respective .inc
file.
; See respective data sheet for additional information on configuration
word.

*****
;
;           Memory Definitions
*****
;
        cblock 0x20
KeyMask      ; PortMask for writing LCD Data
SABitCount   ; Bitcounter for LCWrite
SABuf        ; transmit buffer for LCD
LCParity     ; count '1' bits for parity generation
PortAMask    ; save point for TRISA
PortBMask    ; save point for TRISB
Time         ; count Time interrupts
TimeOut      ; Count Time interrupts until Power OFF
TimeOuth     ; High Byte Count
```

```

RunCMD          ; Samples RunCMDs on every Timer IRQ
wpattern        ; which pattern to be used
Keyold          ; last key press
bcount          ; how many bytes to go
vTest           ; general purpose test
Colour          ; hand over for Colour value
tmp             ; guess what
Pointer:2       ; Pointer into Program Memory to read tables
RGB:3           ; 7bit values for RedGreenBlue
    endc

```

```

w_temp          equ    0x7E    ; variables used for context saving
STATUS_temp     equ    0x7F    ; memory always mapped to bank 0

```

```

;*****
;
;          Bit Definitions
;*****
;
;          Defines
;*****
;

```

```

#define SA_CLK          PORTB,3 ; LC Clock on RB3
                        ; Can't be changed, as PWM is used!!!!!!
                        ; for permanent clock tests

```

```

#define SA_DATA        PORTA    ; Data Lines on RA2, RA3

```

```

#define Key1           02      ; RA2 handles Data of Key1

```

```

#define Key2           03      ; RA3 handles Data of Key1

```

```

#define LED_OFF        0x00

```

```

#define Dark_Red       0x10

```

```

#define Red            0x20

```

```

#define Bright_Red     0x30

```

```

;

```

```

#define Dark_Green     0x04

```

```

#define Green          0x08

```

```

#define Bright_Green   0x0c

```

```

;

```

```

#define Dark_Blue      0x01

```

```

#define Blue           0x02

```

```

#define Bright_Blue    0x03

```

```

#define Dark_White     0x15

```

```

#define White          0x2a

```

```

#define Bright_White   0x3f

```

```

#define SA_CMD_WrDisplay 0x40 ; Set Adr of Display (3 nibbles) and Bitmap
(128, 216, 512 nibbles)

```

```

#define SA_CMD_SetColour 0x41 ; Set Colour awaits one param: '00rrggbb'

```

```

#define SA_CMD_EndTr     0x43 ; end of current command execute

```

```

;#####
    ORG    0x000    ; processor reset vector
    goto   COLD    ; go to beginning of program

    ORG    0x004    ; interrupt vector location
    retfie        ; return from interrupt

COLD
    bcf    STATUS,5    ; set page to 0 (RP0)
    bcf    STATUS,6    ; (RP1)
    call   InitREG    ; Initialise Register
    call   InitHW     ; Initialise HW
; -----
main
    movlw  0x0c
    movwf  KeyMask    ; Key1 and Key2 active

    call   WaitKeyReleased    ; Wait for ....
    clrf  RunCMD      ; deactivate all flagged functions

    movlw  .32        ; wait some time
    call  WAIT        ; to ensure power OK and Keys ready

    incf   Colour, F    ; next Colour
    call   SA_SetColour ; and show it

    movlw  .32        ; wait some time
    call  WAIT        ; as last command already send endofcmd

    bsf   KeyMask, Key1 ; Select Key1
    bcf   KeyMask, Key2 ; DeSelect Key2
    call  SetBitMap6432 ; Setup Address Pointer
    call  TrBitMap     ; Write Bitmap to Keys

    goto  main

;#####
GetData
    movf  Pointer,w    ; Get address byte high
    movwf PCLATH      ; Set Address Byte high to point to Data
    movf  Pointer+1,w  ; get address byte low
    movwf PCL         ; change Program counter to access data
;
IncPointer
    incf  Pointer+1,f  ; increment address byte low
    btfsc STATUS,Z    ; do we need to correct address byte high
    incf  Pointer,f    ; if do it
    return
;

```

```

SetBitMap6432
    movlw    HIGH SA6432          ; get table address byte high
    movwf    Pointer             ; and set Pointer
    movlw    SA6432 & 0xff       ; get table address byte high
    movwf    Pointer+1          ; and set Pointer
    movlw    0x00                ; set Byte Count for SA6432
    movwf    bcount
    return

;
TrBitMap
    bcf     INTCON,7             ; no Interrupts allowed
                                ; in order to ensure proper data transfer
    movlw   SA_CMD_WrDisplay     ; Start WriteDisplay sequence
    call    SAWrite              ; Transmit command
    movlw   0x00                 ; Addressnibble 0: 0
    call    SAWrite              ;
    movlw   0x00                 ; Addressnibble 1: 0
    call    SAWrite              ;
    movlw   0x00                 ; Addressnibble 2: 0
    call    SAWrite              ;

;
Trloop    call    GetData        ; get data of Bitmap
          andlw   0x0f           ; as there is still the lower nibble
          call    SAWrite        ; and send to Key
          call    GetData        ; get same byte again
          andlw   0xf0           ; only 4 bits allowed per transmit -SA
protocol
          movwf   tmp            ; save data
          swapf   tmp,w          ; data has to be in the lower nibble
          call    SAWrite        ; send next nibble of data

          call    IncPointer      ; Point to next Byte of Bitmap
          decfsz  bcount, F      ; Bitmap complete?
          goto    Trloop         ; no - nxt byte
          bsf     INTCON,7       ; allow Interrupts
          return

;*****
;          SAWrite
;*****
;
SAWrite   ; write Data = W to SA Key
          movwf   SABuf         ; save Data to be send in SABuf

          movlw   .8
          movwf   SABitCount    ; set the #bits to 8

bitout
          btfss   SABuf,7       ; check MSB

```

```

        goto      bit_low          ; Skip if Bit was '0'
        bsf      PORTA, Key1      ; Set Data Line of selected Key (1/2) high
        bsf      PORTA, Key2      ; This can be done with both keys even not
selected!
        goto      sndbit
bit_low
        btfscl  KeyMask, Key1    ; if Key1 selected
        bcf      PORTA, Key1      ; send 0-bit to Key1
        btfscl  KeyMask, Key2    ; if Key2 selected
        bcf      PORTA, Key2      ; send 0-bit to Key2
sndbit
        bcf      SA_CLK          ; set clk low data is shifted in now
        bsf      SA_CLK          ; set clk high

        rlf     SABuf, F          ; rotate SABuf right
        ;
        decfsz  SABitCount, F    ; 8 bits done?
        goto    bitout          ; no - nxt bit

        bsf     PORTA, Key1      ; Set Data Line of both Keys high
        bsf     PORTA, Key2      ; in case of auto clock

        retlw   0

        end

```

Wenn Sie weitere Fragen haben, wenden Sie sich bitte an techsupport@e3-keys.com.

HINWEISE

Copyright Hinweis

© 2019-2022 Copyright [E³] Engstler Elektronik Entwicklung GmbH. Alle Rechte vorbehalten.

[E³], The Third Evolution™ und Legacy Mode™ sind Warenzeichen von [E³]. The Keys to Intelligence™ ist eine Marke von I/O Universal Technologies, Inc. und wird mit Genehmigung verwendet. Alle anderen Marken sind Eigentum der jeweiligen Inhaber.

Kein Teil dieser Veröffentlichung darf ohne die ausdrückliche schriftliche Genehmigung der [E³] Engstler Elektronik Entwicklung GmbH kopiert, fotokopiert, reproduziert, übersetzt oder auf ein elektronisches Medium oder eine maschinenlesbare Form reduziert werden.

Technische Hinweise

Dieses Datenblatt wendet sich an technisch qualifiziertes und auf dem Gebiet der Elektronik ausgebildetes Personal.

Die Kenntnis der Elektronik und die technisch korrekte Umsetzung des Inhalts dieses Datenblatts sind Voraussetzung für die problemlose Installation, Inbetriebnahme und den sicheren Betrieb des beschriebenen Produkts. Nur qualifiziertes Personal verfügt über das erforderliche Know-how, um die in diesem Datenblatt gemachten Angaben umzusetzen.

Aus Gründen der Übersichtlichkeit wurden nicht alle Details über das Produkt oder dessen Implementierung, Installation, Betrieb oder Wartung aufgenommen. Sollten Sie zusätzliche Informationen oder weitere Unterstützung benötigen, wenden Sie sich bitte an Ihren örtlichen [E³] Distributor oder an [E³] Engstler Elektronik Entwicklung GmbH unter techsupport@e3-keys.com. Sie können auch unsere Website unter www.e3-keys.com besuchen.

Gewährleistungsausschluss

Die [E³] ENGSTLER ELEKTRONIK ENTWICKLUNG GMBH übernimmt in Bezug auf dieses Datenblatt keine Garantie, weder ausdrücklich noch stillschweigend, und sie haftet nicht für direkte oder indirekte Schäden. In einigen Staaten ist der Ausschluss von beiläufigen Schäden oder Folgeschäden nicht zulässig, so dass diese Erklärung in solchen Fällen möglicherweise nicht gültig ist.

Dieses Datenblatt wurde mit der gebotenen Sorgfalt erstellt. Da jedoch Fehler nicht ausgeschlossen werden können, übernimmt die [E³] Engstler Elektronik Entwicklung

GmbH keine Gewährleistung und keine rechtliche Verantwortung oder Haftung in irgendeiner Form für fehlerhafte Angaben in diesem Datenblatt.

Hinweis zur deutschen Version

Dieses Dokument wurde mit Hilfe von www.DeepL.com/Translator übersetzt. Für den Fall, daß es zwischen dem Originaldokument in Englisch und der deutschen Übersetzung Unterschiede gibt, gilt immer das englische Original als bindend.

ÄNDERUNGSHISTORIE

Version	Datum	Kommentare
1.2	10/31/19	Neues Format
1.3	06/30/20	Neue Formatierung
2.0	06/20/22	Aktualisierte Freigabeversion

[E³] Engstler Elektronik Entwicklung GmbH
Industriering 7 • 63868 Grosswallstadt • Germany

WWW.E3-KEYS.COM