

Description

The serial channel 0 of the μ PD7805x/78005x subseries can be used in 2-wire serial I/O mode, 3-wire serial I/O mode or serial bus interface mode.

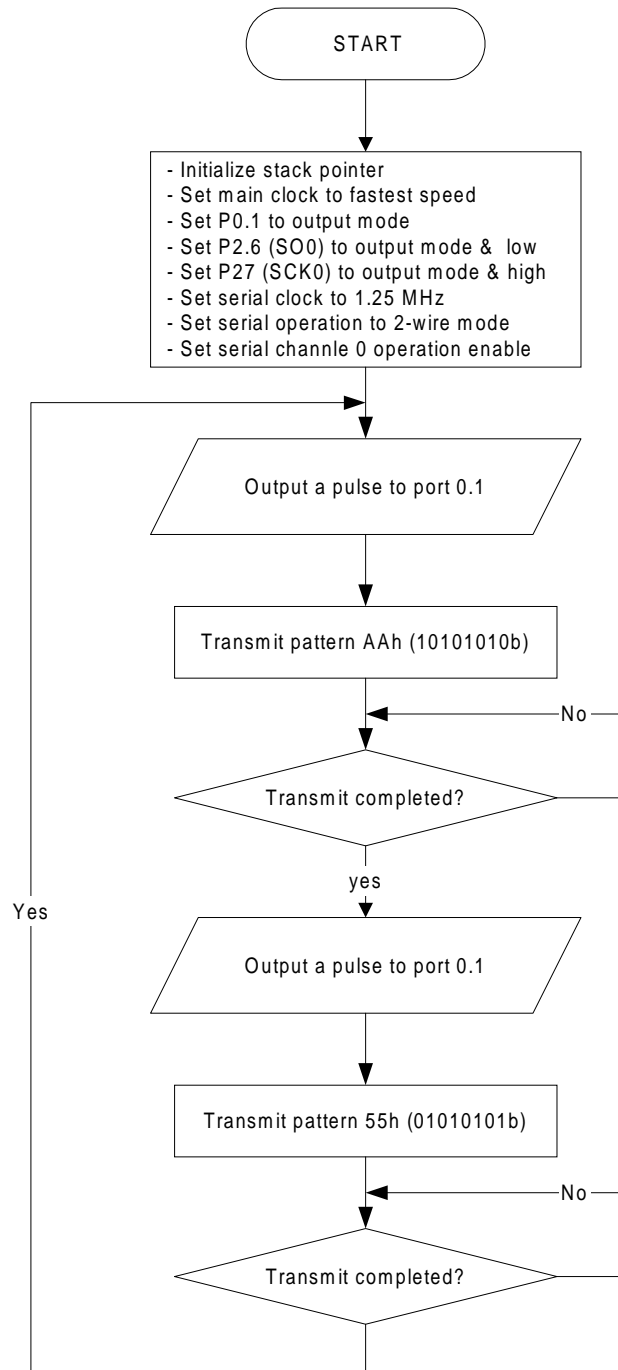
This program demonstrates the 2-wire serial interface in transfer-only mode. A short start pulse is output to port 0.1 before a test pattern is output to the serial output pin SO0/P26.

Caution: When using the program on an NEC emulator, connect pull-up resistors (1 k Ω typ.) to the SO0 data and SCK0 clock signals.

Program Specifications

- Serial clock selection: fx/4 (1.25 MHz)
- Output data format: 8-bit
- Data pattern: AAh followed by 55h continuously
- Pins used in program:
 - Data output: SO0/SB1/P26
 - Serial clock output: SCK0/P27
 - Transmission start pulse: P01/INTP1/TI01

Flowchart



Assembly Language Program

```

;*****
; Date:          08/12/1999
;
; Parameters: - fastest CPU clock
;              (fx=5 MHz; 1 CPU clock cycle = 200 ns)
;              - Serial clock speed:          fx/4 (1.25 MHz)
;              - Data format:                 8-bit, MSB first
;              - Data patterns:               Consecutively AAh and 55H by turn
;              - Data output:                 S00/P26/SB0
;              - Serial clock output:         SCK0/P27
;              - Transmission start signal:   Port 0.1
;*****

;=====
;          Constants/Variables          =
;=====

TXDirPort      equ      PM2.6      ;TX direction port
TXDataPort     equ      P2.6       ;TX Data port
SCKDirPort     equ      PM2.7      ;SCK direction port
SCKDataPort    equ      P2.7       ;SCK data port

;=====
;=          Specify Interrupt vectors   =
;=====

Res_Vec      CSEG AT 0000h          ; Set main program start vector.
              DW          Start

;=====
;          Main Program                =
;=====

MAIN         CSEG
Start:      DI                      ; Disable interrupts
            MOVW      AX, #0FE20h    ; Load SP address
            MOVW      SP, AX         ; Set Stack Pointer
            MOV       OSMS,#01h     ; Don't use scaler
            MOV       PCC, #00h     ; Main system clock at fastest setting
            CLR1     P0.1           ; Latch port 0.1 to low
            CLR1     PM0.1          ; Set port 0.1 to output
            SET1     PUOL.0         ; Use pull-up resistor on port 0.1
            CLR1     TXDirPort      ; Set TX port to output direction
            CLR1     TXDataPort     ; Latch TX port low
            CLR1     SCKDirPort     ; Set SCK0 to output direction
            SET1     SCKDataPort    ; Latch SCK0 high
            MOV      CSIM0,#9Bh     ; Operation settings:
            ;          - Use clock specified in TCL3 register
            ;          - Setup for 2-wire mode
            ;          - Operation enable
            MOV      TCL3,#07h     ; Set serial clock speed to 1.25 MHz
            SET1     CSIMK0         ; Mask serial channel 0 interrupt flag
            CLR1     CSIIF0         ; Clear transfer done interrupt flag
            EI                      ; Enable interrupt
Loop:      SET1     P0.1           ; Set port 0.1

```

```
        CLR1    P0.1           ; Clear port 0.1
        MOV     SIO0,#0AAh     ; Transmit AAh (10101010b)
Wait1:  NOP
        BF      CSIIF0,$Wait1  ; If transmission done, then continue
        CLR1    CSIIF0         ; Clear transmission done interrupt flag
        SET1    P0.1           ; Set port 0.1
        CLR1    P0.1           ; Clear port 0.1
        MOV     SIO0,#055h     ; Transmit 55h (01010101)
Wait2:  NOP
        BF      CSIIF0,$Wait2  ; If transmission done, then continue
        CLR1    CSIIF0         ; Clear transmission done interrupt flag
        BR      Loop           ; Branch back to transmit start
        END
```

C Language Program

```

/*****
; Date:          08/12/1999
;
; Parameters: - fastest CPU clock
;             (fx=5 MHz; 1 CPU clock cycle = 200 ns)
;             - Serial clock speed:          fx/4 (1.25 MHz)
;             - Data format:                 8-bit, MSB first
;             - Data patterns:               Consecutively AAh and 55H by turn
;             - Data output:                 SO0/P26/SB0
;             - Serial clock output:         SCK0/P27
;             - Transmission start signal:   Port 0.1
; *****/
/* extension functions in K0/K0S compiler */
#pragma sfr          /* key word to allow SFR names in C code */
#pragma EI          /* key word for EI instruction in C code */

/*=====
;          Constants/Variables          =
;=====*/
#define TRUE        1
#define FALSE       0
#define TXDirPort   PM2.6      // TX direction port
#define TXDataPort  P2.6       // TX Data port
#define SCKDirPort  PM2.7      // SCK direction port
#define SCKDataPort P2.7       // SCK data port

/*=====
;          Main Program                  =
;=====*/
void main(void)
{
    OSMS = 0x01;          /* Don't use scaler */
    PCC = 0x00;          /* Main system clock at fastest setting */
    P0.1 = 0;           /* Latch port 0.1 to low */
    PM0.1 = 0;          /* Set port 0.1 to output */
    PUOL.0 = 1;         /* Set pull-up resistor to port 0.1 */
    TXDirPort = 0;      /* Set TX to output direction */
    TXDataPort = 0;     /* Latch TX port low */
    SCKDirPort = 0;     /* Set SCK0 to output direction */
    SCKDataPort = 1;    /* Latch SCK0 high */
    CSIM0 = 0x9B;       /* Operation settings:
                        - Use clock specified in TCL3 register
                        - Setup for 2-wire mode
                        - Operation enable */

    TCL3 = 0x07;        /* Set serial clock speed to 1.25 MHz */
    CSIMK0 = 1;         /* Mask serial channel 0 interrupt flag */
    CSIIIF0 = 0;        /* Clear transfer done interrupt flag */
    EI();              /* interrupt enable */
    while(TRUE)        /* while loop */
    {
        P0.1 = 1;      /* Set port 0.1 */
        P0.1 = 0;      /* Clear port 0.1 */
        SIO0 = 0xAA;   /* Transmit AAh (10101010b) */
        while(!CSIIIF0); /* If transmission is done, the continue */
        CSIIIF0 = 0;   /* Clear transmission interrupt flag */
    }
}

```

```

P0.1 = 1;          /* Set port 0.1 */
P0.1 = 0;          /* Clear port 0.1 */
SIO0 = 0x55;       /* Transmit 55h (01010101b) */
while(!CSIIF0);   /* If transmission is done, then continue */
CSIIF0 = 0;        /* Clear transmission done interrupt flag */
    }              /* end of WHILE loop */
}                  /* end of MAIN() */

```

NEC

For literature, call **1-800-366-9782** 7 a.m. to 6 p.m. Pacific time
or FAX your request to **1-800-729-9288**
or visit our web site at **www.necel.com**

In North America: No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics Inc. (NECEL). The information in this document is subject to change without notice. All devices sold by NECEL are covered by the provisions appearing in NECEL Terms and Conditions of Sales only. Including the limitation of liability, warranty, and patent provisions. NECEL makes no warranty, express, statutory, implied or by description, regarding information set forth herein or regarding the freedom of the described devices from patent infringement. NECEL assumes no responsibility for any errors that may appear in this document. NECEL makes no commitments to update or to keep current information contained in this document. The devices listed in this document are not suitable for use in applications such as, but not limited to, aircraft control systems, aerospace equipment, submarine cables, nuclear reactor control systems, and life support systems. "Standard" quality grade devices are recommended for computers, office equipment, communication equipment, test and measurement equipment, machine tools, industrial robots, audio and visual equipment, and other consumer products. For automotive and transportation equipment, traffic control systems, anti-disaster and anti-crime systems, it is recommended that the customer contact the responsible NECEL salesperson to determine the reliability requirements for any such application and any cost adder. NECEL does not recommend or approve use of any of its products in life support devices or systems or in any application where failure could result in injury or death. If customers wish to use NECEL devices in applications not intended by NECEL, customer must contact the responsible NECEL salespeople to determine NECEL's willingness to support a given application.