

HT45B0K USB Bridge Library

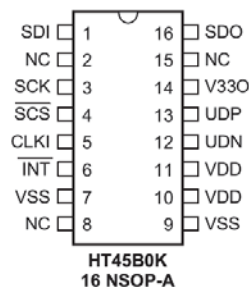
D/N : HA0223E

Introduction

The HT45B0K is an independent USB Bridge IC, that is a peripheral device for microcontrollers. It contains an internal USB 2.0 Full Speed Mode, an SPI interface and an internal 3.3V regulator that can directly use the 5V or 3.3V power supply. The internal PLL uses an external 6/12MHz as the clock source. The HT45B0K contains SPI control lines, \overline{SCS} , SCK, SDI, SDO, and a clock input, CLKI and also interrupt output \overline{INT} and a 3.3V regulator output, and USB control lines, UDP, UDN, VDD and VSS.

Features

- Glueless slave SPI interface to Holtek's MCU
- USB 2.0 full speed module
- 6 USB endpoints
- FIFO : 8, 8, 8, 64, 8, 64 for EP0~EP5 respectively
- Operating voltage: $V_{DD}=3.3V\sim 5.5V$
- 6M/12M (CLKI) external clock input for the USB PLL clock
- Suspend Mode with low suspend current
- Remote Wake-up function
- USB multiple interrupt generation sources
- Access of the corresponding USB FIFO from the USB host
- USB suspend signal from the USB host
- USB wake-up signal from the USB host
- USB reset signal from the USB host
- 16-Pin NSOP package



Operating Principles

The HT45B0K communicates with the USB module via an internal SPI interface to implement the USB transmission.

Registers are Divided into Two Groups: General Registers and FIFO Registers

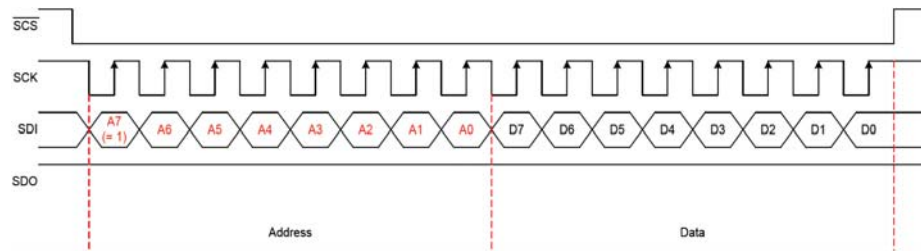
- General Registers: address = USR, USC, UCC, AWR, STALL, SIES, MISC, SETIO, UIC, PIPE, SWRST=00h ~ 0Ch (09h is unused)
- FIFO Registers: address = FIFO0~FIFO5=10h ~ 15h

General Register Operation Format

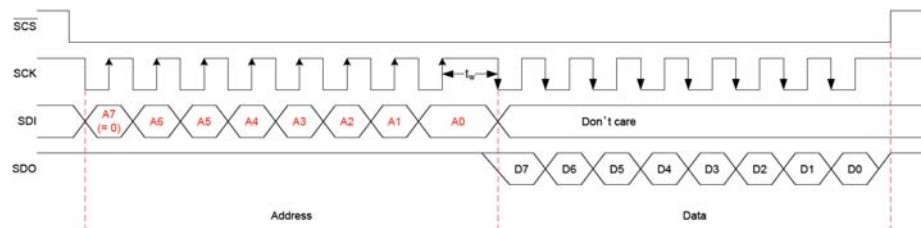
The SPI master asserts the \overline{SCS} line (pulled low) to start the data transaction cycle. When the first 8 data bits are transmitted, the \overline{SCS} signal should not be immediately returned to a high level but instead, must remain at a low level until the complete 16-bit data transaction is completed.

If the \overline{SCS} line is de-asserted (returns high) before a 16-bit data transaction is completed, all data bits will be discarded by the SPI slave.

- General Register Write: 1-byte command/address + 1-byte data



- General Register Read: 1-byte command/address + 1-byte data



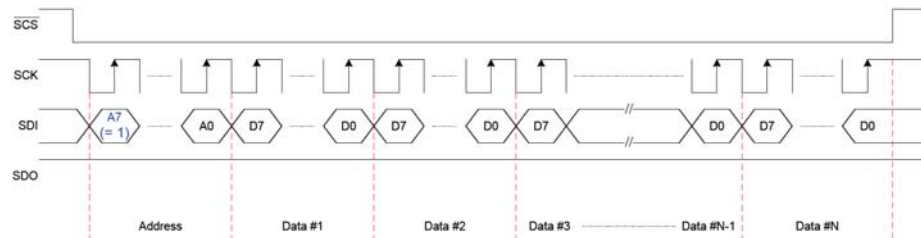
FIFO Register Operation Format

More than 1 byte data can be transmitted or received within one transaction.

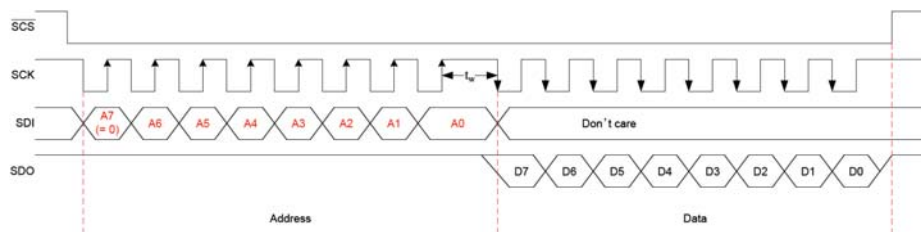
The SPI master asserts the \overline{SCS} line (pulled low) to start the data transaction cycle. The first byte specifies the address of the FIFO registers. The other bytes are data to/from the FIFO.

Once the \overline{SCS} line is de-asserted (pulled high), incomplete data, that is data less than 8 bits, should be discarded.

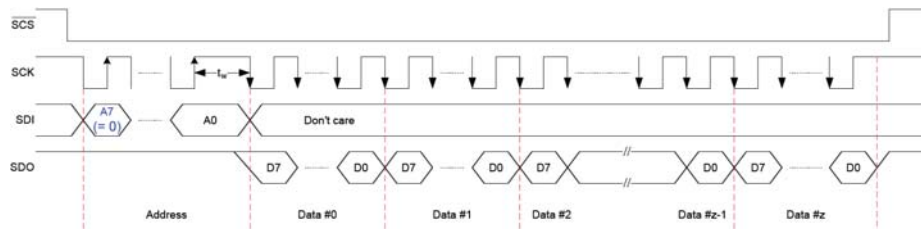
- FIFO Register Write: 1-byte command/address + N-byte data



- FIFO Register Read by single mode (S/C=1): 1-byte command/address + 1-byte data



- FIFO Register Read by continuous mode (S/C=0): 1-byte command/address + N-byte data



All the read and write commands and addresses are combined into one byte. If the highest bit is "0", this means a read command and a "1" means a write command. The A4~A0 bits are the USB register address. To read the value of the USR (address of 01H) register, send 01H before reading the data.

Command Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Read operation	0	S/C	x	A4	A3	A2	A1	A0
Write operation	1	x	x	A4	A3	A2	A1	A0

Note: 1. "x" here stands for "don't care".

2. S/C=0: Read by continuous mode (Default)

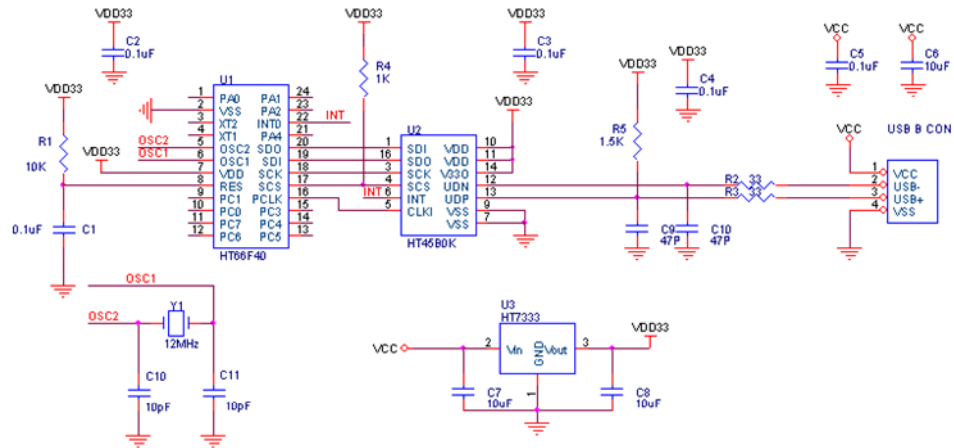
S/C=1: Read by single mode

Read by continuous mode - see the attached reference, the

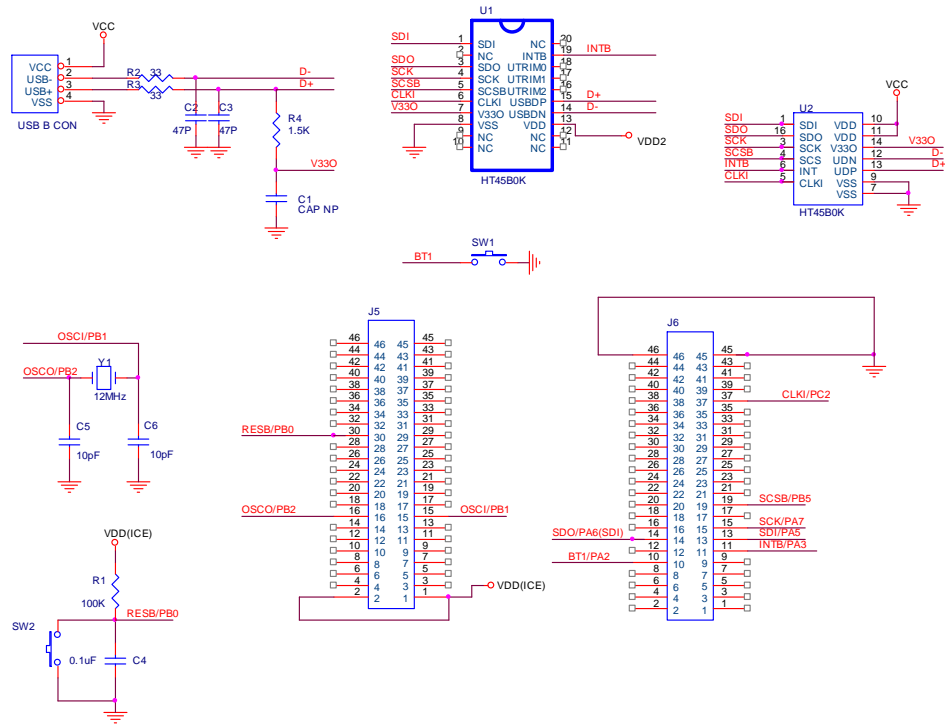
HT66F40test45B0K_continuous_read.zip.

Read by single mode - see the attached reference, HT66F40test45B0K_single_read.zip.

3.3V Application USB Connection

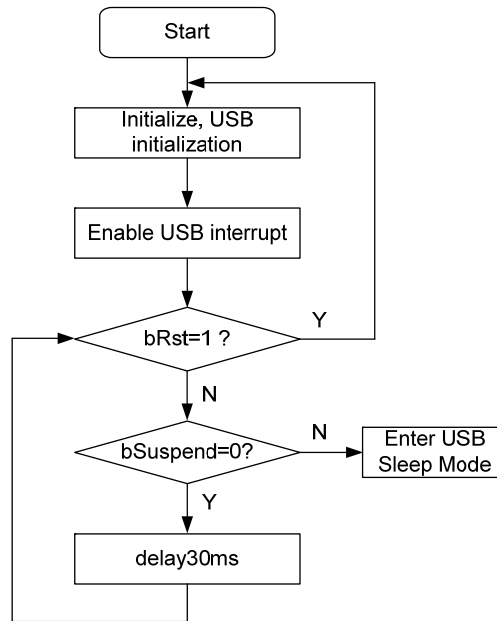


ICE Application USB Connection -- HT66F40 ICE

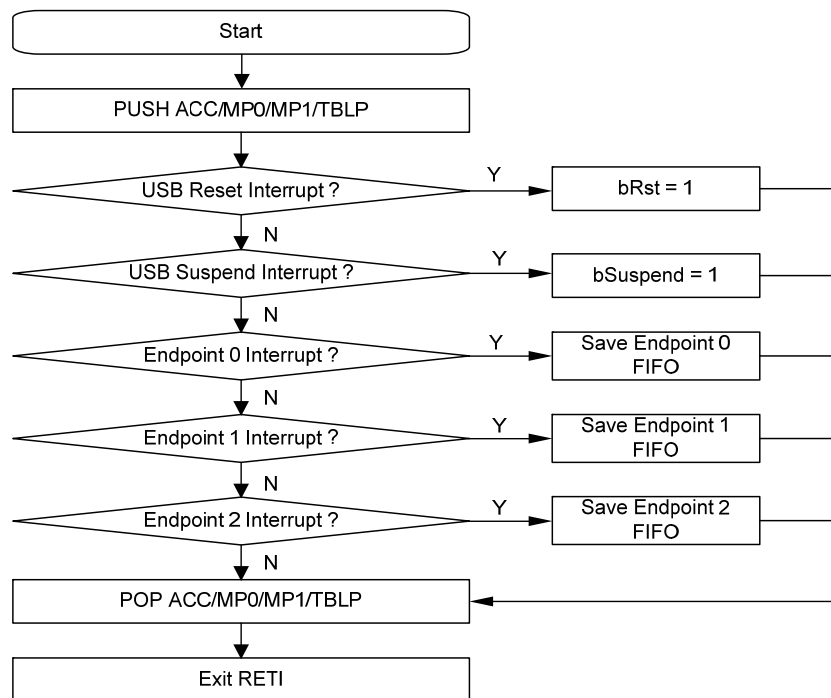


S/W Function Block Flowchart

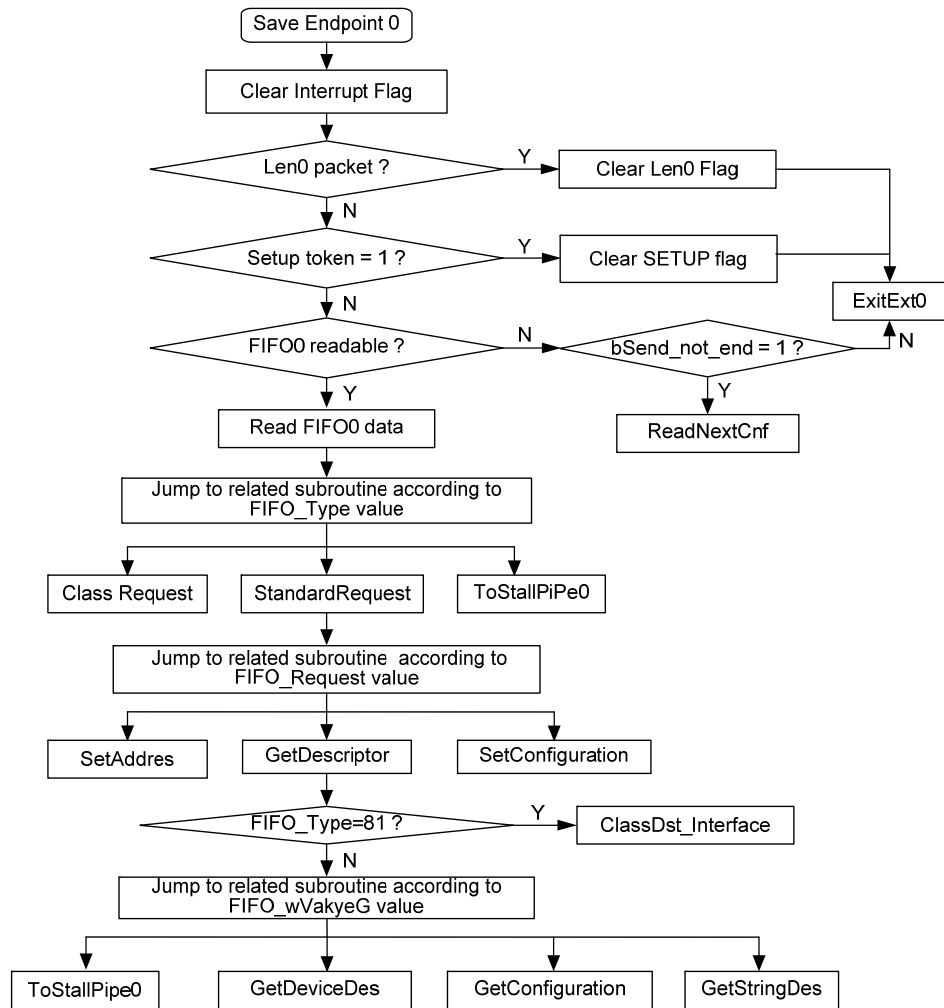
Main Flowchart



USB Interrupt Subroutine Flowchart



Endpoint 0 FIFO Subroutine Flowchart



Appendix Description

- **FUN.INC**
The header file defines all the variables, constants, and macros used in the program.
- **MAIN.ASM**
This ASM file is the register and I/O port initialization routines.
- **USB_INT.ASM**
This ASM file is the USB interrupt service routine where the program will jump to if a USB interrupt occurs.
- **USB_LIB.ASM**
This ASM file is the read/write subroutine to the FIFO.
- **STD.ASM**
This ASM file contains the 11 standard USB driving request subroutines defined by the USB.
- **CLS.ASM**
This ASM file is the request subroutine of the HID components.
- **DES.ASM**
• This ASM file is the definition of the all descriptions.
- **USB_VAR.ASM**
This ASM file defines all the variables that are used by the USB function.
- **USB_BRIDGE.ASM**
This ASM file controls the relevant functions and variable definitions of the HT45B0K.

Revision History

Revision: V1.10

Updated Date: 2010/05/19

Revision History:

The Application Circuit and Associated Files were modified.

Revision: V1.20

Updated Date: 2011/07/26

Revision History:

- The USB 1.1 has been changed to the USB 2.0 Full Speed Mode.
- The Operating Principles section was modified.
- The program code was updated.