

IDENTIFYING VALID EXTENDED CAPABILITIES PORTS IN SMSC FDC37C665/666 DEVICES

INTRODUCTION

Identifying valid Extended Capabilities Ports (ECP) in SMSC FDC37C665/666 devices is a simple, three-step process:

1. Look for a valid SMSC FDC37C665/666 configuration register space;
2. Identify FDC37C665 or FDC37C666 devices;
3. Check for a valid Extended Capabilities Parallel port register space.

SMSC FDC37C665/666 devices can be precisely identified by reading configuration register data directly. All software configuration options available for the FDC37C665 are available for the FDC37C666 except for those options selected by the hardware configuration pins. Since the non-readable options set by hardware configuration in the FDC37C666 include the parallel port address and parallel port mode, initialized ECP port configurations can only be detected indirectly.

C-language code fragments are provided to illustrate all of these operations. For more detail, see the Configuration and Extended Capabilities Parallel Port sections in the FDC37C665/666 data sheets.

IDENTIFICATION PROCESS

To identify FDC37C66X devices, read the configuration registers and look for a valid FDC37C665 or FDC37C666 device-identifier in Register D. The default base address for the 16 configuration registers, also known as the Configuration Select Register (CSR), is 0x3F0. In the FDC37C666, the location of the CSR can be moved to 0x370 via an external programming option.

Functions included below which illustrate the FDC37C66X identification process are:

```
chip check()  
read_reg665()  
read_reg666(int csr)
```

To verify that an ECP port has been initialized, read and write two ECP control registers to detect the presence of an ECP register set. The function below which illustrates this procedure is:

```
valid_ECP_port()
```

C-LANGUAGE DEFINITIONS FOR CODE FRAGMENTS

```
#define CRD 0x0D // Configuration Register D offset
```

Configuration Register D can only be accessed when the FDC is in the Configuration Mode and after the CSR has been initialized to 0x0D. CRD is read only. The default CRD value after power up is 0x65 for the FDC37C665 and 0x66 for the FDC37C666.

```

#define PRIMARY_PORT 0x03F0
    Default address of the FDC37C66X Configuration Select Register.

#define SECONDARY_PORT 0x0370
    FDC37C666-only alternate address for the Configuration Select Register.

#define ECR 0x402 // ECP Extended Control Register
    The ECR register controls the extended ECP parallel port functions. See the Extended Capabilities
    Parallel Port section of the device data sheet for the specific functions of this register.

#define DCR 0x02 // ECP Device Control Register
    The DCR register controls the STROBE, AUTOFD, nINIT, SELECTIN, ACKINTEN, and DIRECTION
    pins on the parallel port. The control register is initialized to zero by the RESET input.

#define DEFAULT_ECR 0x35
    This is the default value for the ECR register. It sets PS/2 Parallel Port Mode, disables the nFAULT
    interrupt, and enables DMA and the DMA service interrupts.

#define DEFAULT_CONTROL 0x15
    This is the default value for the DCR register. See the device data sheet for the status of these
    bits.

```

CODE FRAGMENTS

```

//
// check for SMSC37C66X
// return device number (66, or 65) if detected,
// otherwise return 0
//
int chip_check(void)
{
// check for an FDC37C665 @3F0
read_reg665();
if (config_regs[CRD] == 0x65) return(65);
else // check for an FDC37C666 @3F0
    {read_reg666(PRIMARY_PORT);
    if (config_regs[CRD] == 0x66) return(66);
    else // check for an FDC37C666 @370
        {read_reg666(SECONDARY_PORT);
        if (config_regs[CRD] == 0x66) return(66);}}
return(0);
}

//
// read the 16 SMSC37C665 Super I/O Configuration Registers
// note: the Configuration Select Register (CSR) is the
// PRIMARY_PORT
//
static void read_reg665()
{
int data_port = PRIMARY_PORT + 1, i;

disable(); // disable interrupts
outportb(PRIMARY_PORT, 0x55);
outportb(PRIMARY_PORT, 0x55); // enter 665 configuration mode
enable(); // re-enable interrupts
for (i = 0; i < 16; i++)
    {outportb(PRIMARY_PORT, i); // read config reg 'i'
    config_regs[i] = inportb(data_port);}
outportb(PRIMARY_PORT, 0xAA); // leave configuration mode
}

```

```

//
// read the 16 SMSC37C666 Super I/O Configuration Registers
// note: the Configuration Select Register is the function
// argument
//
static void read_reg666(int csr)
{
int data_port = csr + 1, i;

disable(); // disable interrupts
outportb(csr, 0x44);
outportb(csr, 0x44); // enter 666 configuration mode
enable(); // re-enable interrupts
for (i = 0; i < 16; i++)
    {outportb(csr, i); // read config reg 'i'
    config_regs[i] = inportb(data_port);}
outportb(csr, 0xAA); // leave configuration mode
}

//
// test for a valid ECP port
// return TRUE if port detected
//
int valid_ECP_port(void)
{
int i;
unsigned int base, port_addr[] =
    {0x278, 0x378, 0x3BC, 0x00}; // possible ECP base addresses

for (i = 0; base = port_addr[i]; i++)
    {outportb(base + ECR, DEFAULT_ECR);
    outportb(base + DCR, DEFAULT_CONTROL);
    // See if ECR looks like ECP default values
    if (inportb(base + ECR) == DEFAULT_ECR) return(TRUE);}
return(FALSE); // no ECP port detected
}

```



80 Arkay Drive
Hauppauge, NY 11788
(631) 435-6000
FAX (631) 273-3123

Copyright © SMSC 2004. All rights reserved.

Circuit diagrams and other information relating to SMSC products are included as a means of illustrating typical applications. Consequently, complete information sufficient for construction purposes is not necessarily given. Although the information has been checked and is believed to be accurate, no responsibility is assumed for inaccuracies. SMSC reserves the right to make changes to specifications and product descriptions at any time without notice. Contact your local SMSC sales office to obtain the latest specifications before placing your product order. The provision of this information does not convey to the purchaser of the described semiconductor devices any licenses under any patent rights or other intellectual property rights of SMSC or others. All sales are expressly conditional on your agreement to the terms and conditions of the most recently dated version of SMSC's standard Terms of Sale Agreement dated before the date of your order (the "Terms of Sale Agreement"). The product may contain design defects or errors known as anomalies which may cause the product's functions to deviate from published specifications. Anomaly sheets are available upon request. SMSC products are not designed, intended, authorized or warranted for use in any life support or other application where product failure could cause or contribute to personal injury or severe property damage. Any and all such uses without prior written approval of an Officer of SMSC and further testing and/or modification will be fully at the risk of the customer. Copies of this document or other SMSC literature, as well as the Terms of Sale Agreement, may be obtained by visiting SMSC's website at <http://www.smsc.com>. SMSC is a registered trademark of Standard Microsystems Corporation ("SMSC"). Product names and company names are the trademarks of their respective holders.

SMSC DISCLAIMS AND EXCLUDES ANY AND ALL WARRANTIES, INCLUDING WITHOUT LIMITATION ANY AND ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND AGAINST INFRINGEMENT AND THE LIKE, AND ANY AND ALL WARRANTIES ARISING FROM ANY COURSE OF DEALING OR USAGE OF TRADE.

IN NO EVENT SHALL SMSC BE LIABLE FOR ANY DIRECT, INCIDENTAL, INDIRECT, SPECIAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES; OR FOR LOST DATA, PROFITS, SAVINGS OR REVENUES OF ANY KIND; REGARDLESS OF THE FORM OF ACTION, WHETHER BASED ON CONTRACT; TORT; NEGLIGENCE OF SMSC OR OTHERS; STRICT LIABILITY; BREACH OF WARRANTY; OR OTHERWISE; WHETHER OR NOT ANY REMEDY OF BUYER IS HELD TO HAVE FAILED OF ITS ESSENTIAL PURPOSE, AND WHETHER OR NOT SMSC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.