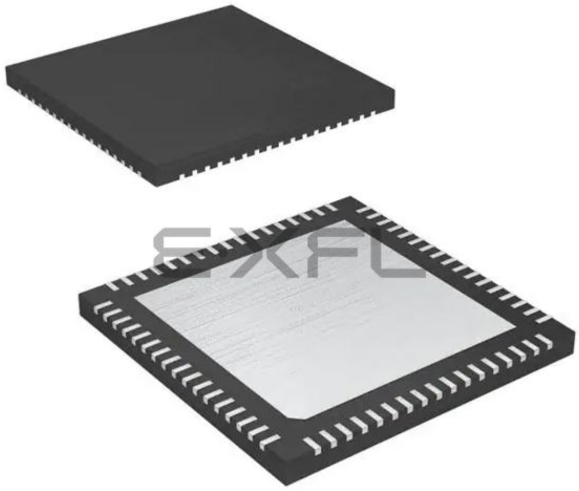
Analog Devices Inc./Maxim Integrated - 73S1217F-68IM/F/P Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Discontinued at Digi-Key
Core Processor	80515
Core Size	8-Bit
Speed	24MHz
Connectivity	I²C, SmartCard, UART/USART, USB
Peripherals	LED, POR, WDT
Number of I/O	8
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 6.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	68-VFQFN Exposed Pad
Supplier Device Package	68-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/73s1217f-68im-f-p

Table of Contents

1.7.5 Interrupts

The 80515 core provides 10 interrupt sources with four priority levels. Each source has its own request flag(s) located in a special function register (TCON, IRCON, and SCON). Each interrupt requested by the corresponding flag can be individually enabled or disabled by the enable bits in SFRs IEN0, IEN1, and IEN2. Some of the 10 sources are multiplexed in order to expand the number of interrupt sources. These will be described in more detail in the respective sections.

External interrupts are the interrupts external to the 80515 core, i.e. signals that originate in other parts of the 73S1217F, for example the USB interface, USR I/O, RTC, smart card interface, analog comparators, etc. The external interrupt configuration is shown in Figure 9.

.

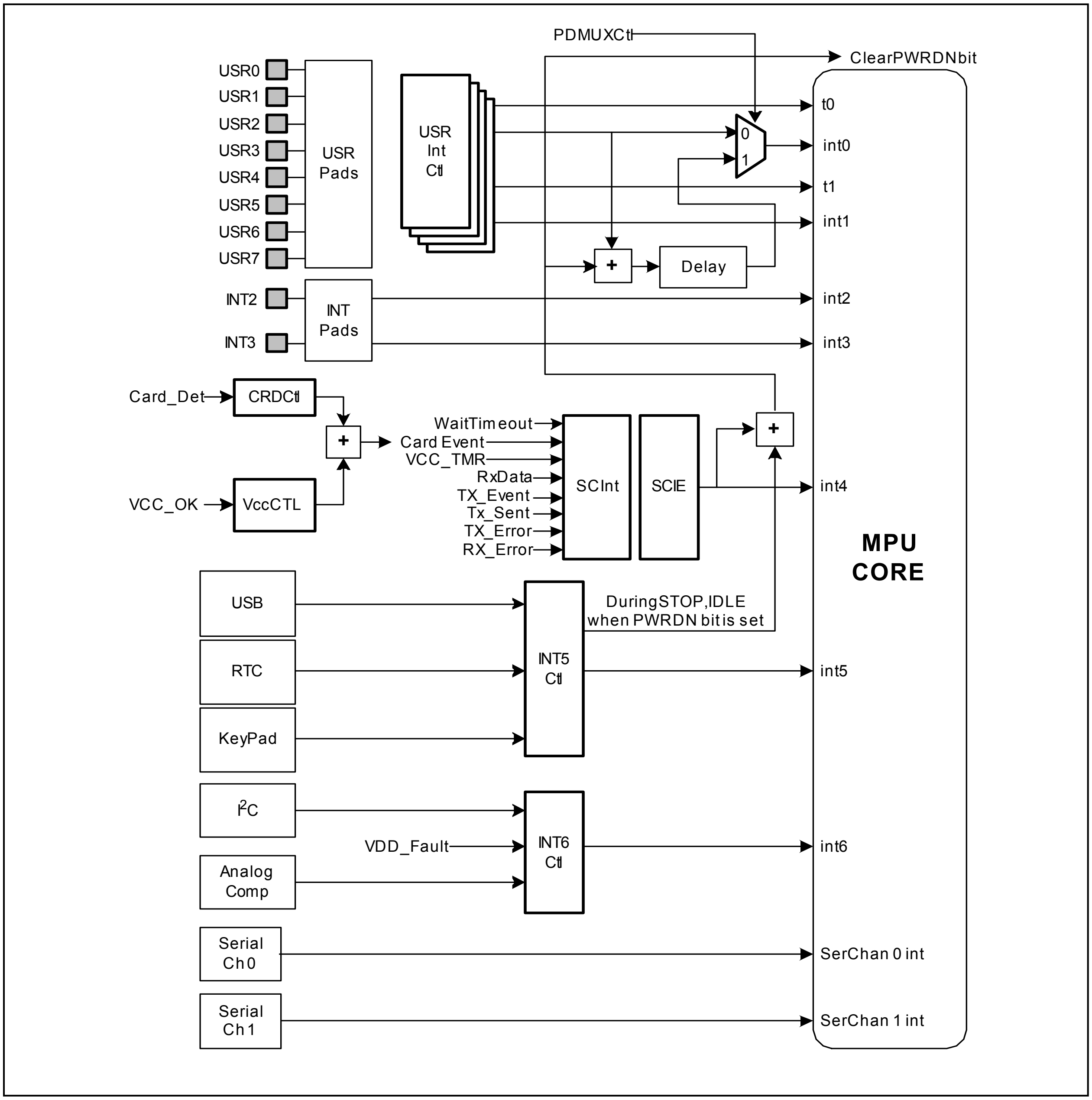


Figure 9: External Interrupt Configuration

1.7.5.1 Interrupt Overview

When an interrupt occurs, the MPU will vector to the predetermined address as shown in Table 32. Once the interrupt service has begun, it can only be interrupted by a higher priority interrupt. The interrupt service is terminated by a return from the RETI instruction. When an RETI is performed, the processor will return to the instruction that would have been next when the interrupt occurred.

When the interrupt condition occurs, the processor will also indicate this by setting a flag bit. This bit is set regardless of whether the interrupt is enabled or disabled. Each interrupt flag is sampled once per machine cycle, then samples are polled by the hardware. If the sample indicates a pending interrupt when the interrupt is enabled, then the interrupt request flag is set. On the next instruction cycle, the interrupt will be acknowledged by hardware forcing an LCALL to the appropriate vector address.

Interrupt response will require a varying amount of time depending on the state of the MPU when the interrupt occurs. If the MPU is performing an interrupt service with equal or greater priority, the new interrupt will not be invoked. In other cases, the response time depends on the current instruction. The fastest possible response to an interrupt is 7 machine cycles. This includes one machine cycle for detecting the interrupt and six cycles to perform the LCALL.

1.7.5.2 Special Function Registers for Interrupts

Interrupt Enable 0 Register (IEN0): 0xA8 ← 0x00

Table 19: The IEN0 Register

MSB							LSB
EAL	WDT	–	ES0	ET1	EX1	ET0	EX0

Bit	Symbol	Function
IEN0.7	EAL	EAL = 0 – disable all interrupts.
IEN0.6	WDT	Not used for interrupt control.
IEN0.5	–	
IEN0.4	ES0	ES0 = 0 – disable serial channel 0 interrupt.
IEN0.3	ET1	ET1 = 0 – disable timer 1 overflow interrupt.
IEN0.2	EX1	EX1 = 0 – disable external interrupt 1.
IEN0.1	ET0	ET0 = 0 – disable timer 0 overflow interrupt.
IEN0.0	EX0	EX0 = 0 – disable external interrupt 0.

1.7.6 UART

The 80515 core of the 73S1217F includes two separate UARTs that can be programmed to communicate with a host. The 73S1217F can only connect one UART at a time since there is only one set of TX and Rx pins. The MISCtl0 register is used to select which UART is connected to the TX and RX pins. Each UART has a different set of operating modes that the user can select according to their needs. The UART is a dedicated 2-wire serial interface, which can communicate with an external host processor at up to 115,200 bits/s. The TX and RX pins operate at the V_{DD} supply voltage levels and should never exceed 3.6V. The operation of each pin is as follows:

RX: Serial input data is applied at this pin. Conforming to RS-232 standard, the bytes are input LSB first. The voltage applied at RX must not exceed 3.6V.

TX: This pin is used to output the serial data. The bytes are output LSB first.

The 73S1217F has several UART-related read/write registers. All UART transfers are programmable for parity enable, parity select, 2 stop bits/1 stop bit and XON/XOFF options for variable communication baud rates from 300 to 115200 bps. Table 47 shows the selectable UART operation modes and Table 48 shows how the baud rates are calculated.

Table 33: UART Modes

	UART 0	**UART 1**
Mode 0	N/A	Start bit, 8 data bits, parity, stop bit, variable baud rate (internal baud rate generator)
Mode 1	Start bit, 8 data bits, stop bit, variable baud rate (internal baud rate generator or timer 1)	Start bit, 8 data bits, stop bit, variable baud rate (internal baud rate generator)
Mode 2	Start bit, 8 data bits, parity, stop bit, fixed baud rate 1/32 or 1/64 of f_{CKMPU}	N/A
Mode 3	Start bit, 8 data bits, parity, stop bit, variable baud rate (internal baud rate generator or timer 1)	N/A

Note: Parity of serial data is available through the P flag of the accumulator. Seven-bit serial modes with parity, such as those used by the FLAG protocol, can be simulated by setting and reading bit 7 of 8-bit output data. Seven-bit serial modes without parity can be simulated by setting bit 7 to a constant 1.8-bit serial modes with parity can be simulated by setting and reading the 9th bit, using the control bits S0CON3 and S1CON3 in the S0CON and S1CON SFRs.

Table 34: Baud Rate Generation

	Using Timer 1	**Using Internal Baud Rate Generator**
Serial Interface 0	$2^{smod} * f_{CKMPU}/ (384 * (256\text{-}TH1))$	$2^{smod} * f_{CKMPU}/(64 * (2^{10}\text{-}S0REL))$
Serial Interface 1	N/A	$f_{CKMPU}/(32 * (2^{10}\text{-}S1REL))$

Note: S0REL (9:0) and S1REL (9:0) are 10-bit values derived by combining bits from the respective timer reload registers SxRELH (bits 1:0) and SxRELL (bits 7:0). TH1 is the high byte of timer 1. The SMOD bit is located in the PCON SFR.

Miscellaneous Control Register 0 (MISCtl0): 0xFFF1 ← 0x00

Transmit and receive (TX and RX) pin selection and loop back test configuration are set up via this register.

Table 37: The MISCtl0 Register

MSB							LSB
PWRDN	–	–	–	–	–	SLPBK	SSEL

Bit	Symbol	Function
MISCtl0.7	PWRDN	This bit places the 73S1217F into a power down state.
MISCtl0.6	–	
MISCtl0.5	–	
MISCtl0.4	–	
MISCtl0.3	–	
MISCtl0.2	–	
MISCtl0.1	SLPBK	1 = UART loop back testing mode. The pins TXD and RXD are to be connected together externally (with SLPBK =1) and therefore: SLPBK SSEL Mode 0 0 normal using Serial_0 0 1 normal using Serial_1 1 0 Serial_0 TX feeds Serial_1 RX 1 1 Serial_1 TX feeds Serial_0 RX
MISCtl0.0	SSEL	Selects either Serial_1 if set =1 or Serial_0 if set = 0 to be connected to RXD and TXD pins.

1.7.6.1 Serial Interface 0

The Serial Interface 0 can operate in four modes:

- **Mode 0**
  Pin RX serves as input and output. TX outputs the shift clock. Eight bits are transmitted with the LSB first. The baud rate is fixed at 1/12 of the crystal frequency. Reception is initialized in Mode 0 by setting the flags in S0CON as follows: RI0 = 0 and REN0 = 1. In other modes, a start bit when REN0 = 1 starts receiving serial data.
- **Mode 1**
  Pin RX serves as input, and TX serves as serial output. No external shift clock is used, 10 bits are transmitted: a start bit (always 0), 8 data bits (LSB first), and a stop bit (always 1). On receive, a start bit synchronizes the transmission, 8 data bits are available by reading S0BUF, and stop bit sets the flag RB80 in the Special Function Register S0CON. In mode 1 either internal baud rate generator or timer 1 can be use to specify baud rate.
- **Mode 2**
  This mode is similar to Mode 1, with two differences. The baud rate is fixed at 1/32 or 1/64 of oscillator frequency and 11 bits are transmitted or received: a start bit (0), 8 data bits (LSB first), a programmable 9th bit, and a stop bit (1). The 9th bit can be used to control the parity of the serial interface: at transmission, bit TB80 in S0CON is output as the 9th bit, and at receive, the 9th bit affects RB80 in Special Function Register S0CON.
- **Mode 3**
  The only difference between Mode 2 and Mode 3 is that in Mode 3 either internal baud rate generator or timer 1 can be use to specify baud rate.

The S0BUF register is used to read/write data to/from the serial 0 interface.

Serial Interface 0 Control Register (S0CON): 0x9B ← 0x00

1.7.7 Timers and Counters

The 80515 has two 16-bit timer/counter registers: Timer 0 and Timer 1. These registers can be configured for counter or timer operations.

In timer mode, the register is incremented every machine cycle, meaning that it counts up after every 12 periods of the MPU clock signal.

In counter mode, the register is incremented when the falling edge is observed at the corresponding input signal T0 or T1 (T0 and T1 are the timer gating inputs derived from USR[0:7] pins, see the User (USR) Ports section). Since it takes 2 machine cycles to recognize a 1-to-0 event, the maximum input count rate is 1/2 of the oscillator frequency. There are no restrictions on the duty cycle, however to ensure proper recognition of 0 or 1 state, an input should be stable for at least 1 machine cycle.

Four operating modes can be selected for Timer 0 and Timer 1. Two Special Function Registers (TMOD and TCON) are used to select the appropriate mode.

The Timer 0 load registers are designated as TL0 and TH0 and the Timer 1 load registers are designated as TL1 and TH1.

Timer/Counter Mode Control Register (TMOD): 0x89 ← 0x00

Table 40: The TMOD Register

MSB							LSB
GATE	C/T	M1	M0	GATE	C/T	M1	M0
Timer 1				**Timer 0**			

Bits TR1 and TR0 in the TCON register start their associated timers when set.

Bit	Symbol	Function
TMOD.7 TMOD.3	GATE	If set, enables external gate control (USR pin(s) connected to T0 or T1 for Counter 0 or 1, respectively). When T0 or T1 is high, and TRx bit is set (see the TCON register), a counter is incremented every falling edge on T0 or T1 input pin. If not set, the TRx bit controls the corresponding timer.
TMOD.6 TMOD.2	C/T	Selects Timer or Counter operation. When set to 1, the counter operation is performed based on the falling edge of T0 or T1. When cleared to 0, the corresponding register will function as a timer.
TMOD.5 TMOD.1	M1	Selects the mode for Timer/Counter 0 or Timer/Counter 1, as shown in TMOD description.
TMOD.4 TMOD.0	M0	Selects the mode for Timer/Counter 0 or Timer/Counter 1, as shown in TMOD description.

External Interrupt Control Register (USRIntCtl1) : 0xFF90 ← 0x00

Table 50: The USRIntCtl1 Register

MSB							LSB
–	U1IS.6	U1IS.5	U1IS.4	–	U0IS.2	U0IS.1	U0IS.0

External Interrupt Control Register (USRIntCtl2) : 0xFF91 ← 0x00

Table 51: The USRIntCtl2 Register

MSB							LSB
–	U3IS.6	U3IS.5	U3IS.4	–	U2IS.2	U2IS.1	U2IS.0

External Interrupt Control Register (USRIntCtl3) : 0xFF92 ← 0x00

Table 52: The USRIntCtl3 Register

MSB							LSB
–	U5IS.6	U5IS.5	U5IS.4	–	U4IS.2	U4IS.1	U4IS.0

External Interrupt Control Register (USRIntCtl4) : 0xFF93 ← 0x00

Table 53: The USRIntCtl4 Register

MSB							LSB
–	U7IS.6	U7IS.5	U7IS.4	–	U6IS.2	U6IS.1	U6IS.0

Device Address Register (DAR): 0xFF80 ← 0x00

Table 62: The DAR Register

MSB LSB

DVADR.6	DVADR.5	DVADR.4	DVADR.3	DVADR.2	DVADR.1	DVADR.0	I2CRW

Bit	Symbol	Function
DAR.7	DVADR [0:6]	Slave device address.
DAR.6		
DAR.5		
DAR.4		
DAR.3		
DAR.2		
DAR.1		
DAR.0	I2CRW	If set = 0, the transaction is a write operation. If set = 1, read.

I2C Write Data Register (WDR): 0XFF81 ← 0x00

Table 63: The WDR Register

MSB LSB

WDR.7	WDR.6	WDR.5	WDR.4	WDR.3	WDR.2	WDR.1	WDR.0

Bit	Function
WDR.7	Data to be written to the I^{2}C slave device.
WDR.6	
WDR.5	
WDR.4	
WDR.3	
WDR.2	
WDR.1	
WDR.0	

column scanning at an adjustable scanning rate and column scanning order through registers KSCAN and KORDERL / KORDERH. Key scanning is disabled at reset and must be enabled by firmware. When a valid key is detected, an interrupt is generated and the valid value of the pressed key is automatically written into the KCOL and KROW registers. The keypad interface uses a 1kHz clock derived from either the 32768Hz crystal or the 12MHz crystal. The selection of the clock source is made external to this block, by setting bit 3 – 32KBEN – in the MCLKCtl register, see the oscillator and clock generation section). Disabling the 32kHz oscillator will source the 1kHz clock from the 12MHz main oscillator and divide it down. Setting bit 6 – KBEN – in the MCLKCtl register will enable keypad scanning and debouncing. The keypad size can be adjusted within the KSIZE register.

Normal scanning is performed by hardware when the bit SCNEN is set at 1 in the KSTAT register. Figure 14 shows the flowchart of how the hardware scanning operates. In order to minimize power, scanning does not occur until a key-press is detected. Once hardware key scanning is enabled, the hardware drives all column outputs low and waits for a low to be detected on one of the inputs. When a low is detected on any row, and before key scanning starts, the hardware checks that the low level is still detected after a debounce time. The debounce time is defined by firmware in the KSCAN register (bits 7:0, DBTIME). Debounce times from 4ms to 256ms in 4ms increments are supported. If a key is not pressed after the debounce time, the hardware will go back to looking for any input to be low. If a key is confirmed to be pressed, key scanning begins.

Key scanning asserts one of the 5 drive lines (COL 4:0) low and looks for a low on a sense line indicating that a key is pressed at the intersection of the drive/sense line in the keypad. After all sense lines have been checked without a key-press being detected, the next column line is asserted. The time between checking each sense line is the scan time and is defined by firmware in the KSCAN register (bits 0:1 – SCTIME). Scan times from 1ms to 4ms are supported. Scanning order does not affect the scan time. This scanning continues until the entire keypad is scanned. If only one key is pressed, a valid key is detected. Simultaneous key presses are not considered as valid (If two keys are pressed, no key is reported to firmware).

Possible scrambling of the column scan order is provided by means of the KORDERL and KORDERH registers that define the order of column scanning. Values in these registers must be updated every time a new keyboard scan order is desired. It is not possible to change the order of scanning the sense lines. The column and row intersection for the detected valid key are stored in the KCOL and KROW registers. When a valid key is detected, an interrupt is generated. Firmware can then read those registers to determine which key had been pressed. After reading the KCOL and KROW registers, the firmware can update the KORDERL / KORDERH registers if a new scan order is needed.

When the SCNEN bit is enabled in the KSTAT register, the KCOL and KROW registers are only updated after a valid key has been identified. The hardware does not wait for the firmware to service the interrupt in order to proceed with the key scanning process. Once the valid key (or invalid key – e.g. two keys pressed) is detected, the hardware waits for the key to be released. Once the key is released, the debounce timer is started. If the key is not still released after the debounce time, the debounce counter starts again. After a key release, all columns will be driven low as before and the process will repeat waiting for any key to be pressed.

When the SCNEN bit is disabled, all drive outputs are set to the value in the KCOL register. If firmware clears the SCNEN bit in the middle of a key scan, the KCOL register contains the last value stored in there which will then be reflected on the output pins.

A bypass mode is provided so that the firmware can do the key scanning manually (SCNEN bit must be cleared). In bypass mode, the firmware writes/reads the Column and Row registers to perform the key scanning.

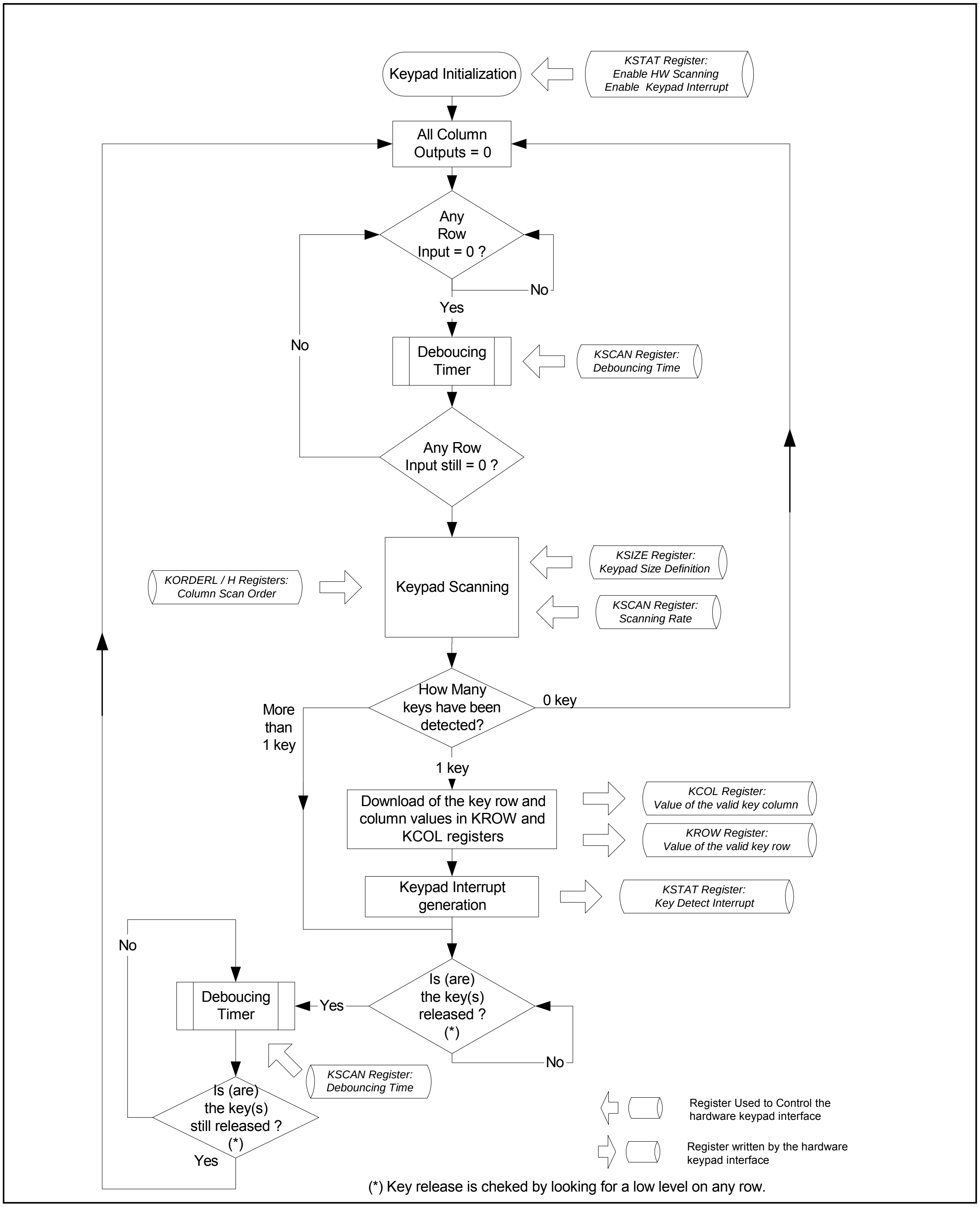
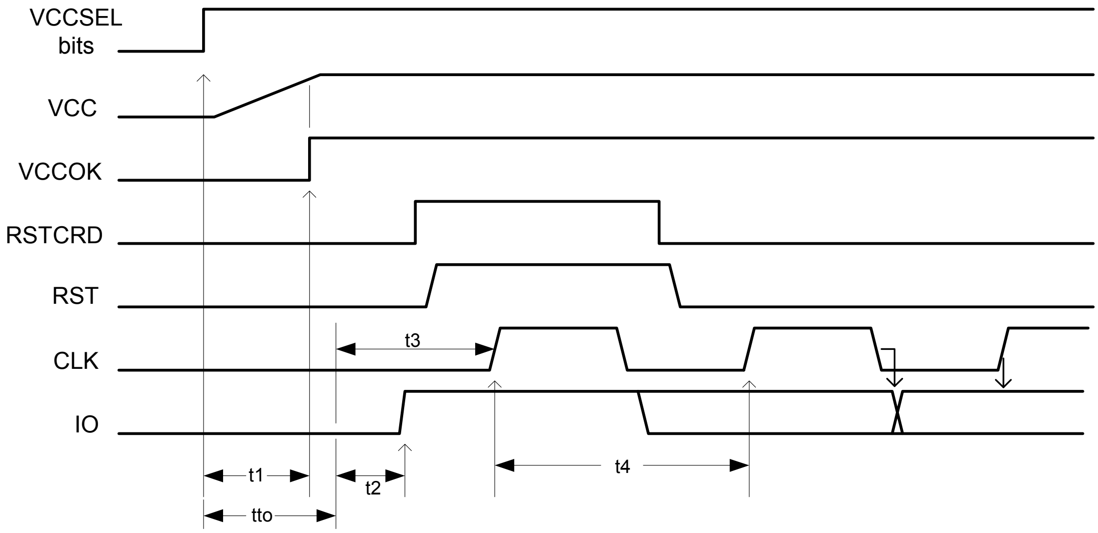

Keypad Initialization
KSTAT Register:
Enable HW Scanning
Enable Keypad Interrupt
All Column Outputs = 0
Any Row Input = 0 ?
No
Yes
No
Deboucing Timer
KSCAN Register:
Debouncing Time
Any Row Input still = 0 ?
KORDERL / H Registers:
Column Scan Order
Keypad Scanning
KSIZE Register:
Keypad Size Definition
KSCAN Register:
Scanning Rate
How Many keys have been detected?
More than 1 key
0 key
1 key
Download of the key row and column values in KROW and KCOL registers
KCOL Register:
Value of the valid key column
KROW Register:
Value of the valid key row
Keypad Interrupt generation
KSTAT Register:
Key Detect Interrupt
No
Deboucing Timer
Yes
Is (are) the key(s) released ? (*)
No
KSCAN Register:
Debouncing Time
Is (are) the key(s) still released ? (*)
Yes
Register Used to Control the hardware keypad interface
Register written by the hardware keypad interface
(*) Key release is cheked by looking for a low level on any row.

Figure 14: Keypad Interface Flow Chart

When the SCISYN or SCESNC bits (SPrtcol, bit 7, bit 5, respectively) are set, the selected smart card interface operates in synchronous mode and there are changes in the definition and behavior of pertinent register bits and associated circuitry. The following requirements are to be noted:

1. The source for the smart card clock (CLK or SCLK) is the ETU counter. Only the actively selected interface can have a running synchronous clock. In contrast, an unselected interface may have a running clock in the asynchronous mode of operation.
2. The control bits CLKLVL, SCLKLVL, CLKOFF, and SCLKOFF are functional in synchronous mode. When the CLKOFF bit is set, it will not truncate either the logic low or logic high period when the (stop at) level is of opposite polarity. The CLK/SCLK signal will complete a correct logic low or logic high duty cycle before stopping at the selected level. The CLK “start” is a result of the falling edge of the CLKOFF bit. Setting clock to run when it is stopped low will result in a half period of low before going high. Setting clock to run when it is stopped high will result in the clock going low immediately and then running at the selected rate with 50% duty cycle (within the limitations of the ETU divisor value).
3. The Rlen(7:0) is configured to count the falling edges of the ETU clock (CLK or SCLK) after it has been loaded with a value from 1 to 255. A value of 0 disables the counting function and RLen functions such as I/O source selection (I/O signal bypasses the FIFOs and is controlled by the SCCLK/SCECLK SFRs). When the RLen counter reaches the “max” (loaded) value, it sets the WAITTO interrupt (SCInt, bit 7), which is maskable via WTOIEN (SCIE, bit 7). I2CMODEIt must be reloaded in order to start the counting/clocking process again. This allows the processor to select the number of CLK cycles and hence, the number of bits to be read or written to/from the card.
4. The FIFO is not clocked by the first CLK (falling) edge resulting from a CLKOFF de-assertion (a clock start event) when the CLK was stopped in the high state and RLen has been loaded but not yet clocked.
5. The state of the pin IO or SIO is sampled on the rising edge of CLK/SCLK and stored in bit 5 of the SCCtl/SCECtl register.
6. When Rlen = max or 0 and I2CMODE = 1 (STXCtl, b7), the IO or SIO signal is directly controlled by the data and direction bits in the respective SCCtl and SCECtl register. The state of the data in the TX FIFO is bypassed.
7. In the SPrtcol register, bit 6 (MODE9/8B) becomes active. When set, the RXData FIFO will read nine-bit words with the state of the ninth bit being readable in SRXCtl, bit 7 (B9DAT). The RXDAV interrupt will occur when the ninth bit has been clocked in (rising edge of CLK or SCLK).
8. Care must be taken to clear the RX and TX FIFOs at the start of any transaction. The user shall read the RX FIFO until it indicates empty status. Reading the TX FIFO twice will reset the input byte pointer and the next write to the TX FIFO will load the byte to the “first out” position. Note that the bit pointer (serializer/deserializer) is reset to bit 0 on any change of the TX/RXD bit.

Special bits that are only active for sync mode include: SRXCtl, b7 “BIT9DAT”, SPrtcol b6 “MODE9/8B”, STXCtl, b7 “I2CMODE”, and the definition of SCInt b7, which was “WAITTO”, becomes RLenINT interrupt, and SCIE b7, which was “WTOIEN”, becomes RLenIEN.



t1: The time from setting VCCSEL bits until VCCOK = 1.

tto: The time from setting VCCSEL bits until VCCTMR times out. At t1 (if RDYST = 1) or tto (if RDYST = 0), activation starts. It is suggested to have RDYST = 0 and use the VCCTMR interrupt to let MPU know when sequence is starting.

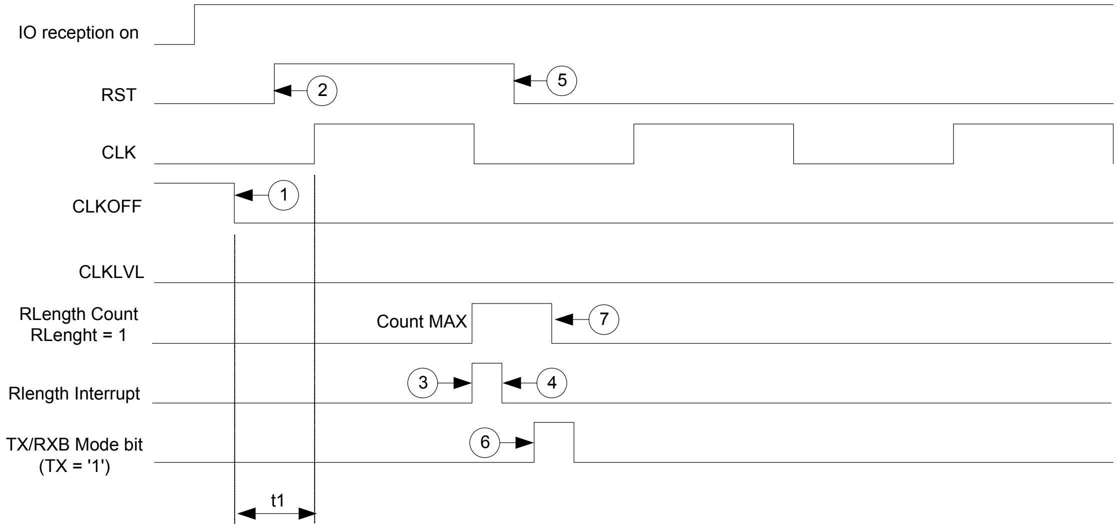
t2: time from start of activation (no external indication) until IO goes into reception mode (= 1). This is approximately 4 SCCLK (or SCECLK) clock cycles.

t3: minimum one half of ETU period.

t4: ETU period.

Note that in Sync mode, IO as input is sampled on the rising edge of CLK. IO changes on the falling edge of CLK, either from the card or from the 73S1217F. The RST signal to the card is directly controlled by the RSTCRD bit (non-inverted) via the MPU and is shown as an example of a possible RST pattern.

Figure 22: Synchronous Activation



1. Clear CLKOFF after Card is in reception mode.
2. Set RST bit.
3. Interrupt is generated when Rlength counter is MAX.
4. Read and clear Interrupt.
5. Clear RST bit.
6. Toggle TX/RXB to reset bit counter.
7. Reload RLength Counter.

t1. CLK wll start at least 1/2 ETU after CLKOFF is set low when CLKLVL = 0

Figure 23: Example of Sync Mode Operation: Generating/Reading ATR Signals

Smart Card Interrupt Register (SCInt): 0xFE01 ← 0x00

When the smart card interrupt is asserted, the firmware can read this register to determine the actual cause of the interrupt. The bits are cleared when this register is read. Each interrupt can be disabled by the Smart Card Interrupt Enable register. Error processing must be handled by the firmware. This register relates to the interface that is active – see the SCSel register (above).

Table 80: The SCInt Register

MSB							LSB
WAITTO	CRDEVT	VCCTMRI	RXDAV	TXEVT	TXSENT	TXERR	RXERR

Bit	Symbol	Function
SCInt.7	WAITTO	Wait Timeout – An ATR or card wait timeout has occurred. In sync mode, this interrupt is asserted when the RLen counter (it advances on falling edges of CLK/ETU) reaches the loaded (max) value. This bit is cleared when the SCINT register is read. When running in Synchronous Clock Stop Mode, this bit becomes RLenINT interrupt (set when the Rlen counter reaches the terminal count).
SCInt.6	CRDEVT	Card Event – A card event is signaled via pin DETCARD either when the Card was inserted or removed (read the CRDCtl register to determine card presence) or there was a fault condition in the interface circuitry. This bit is functional even if the smart card logic clock is disabled and when the PWRDN bit is set. This bit is cleared when the SCInt register is read.
SCInt.5	VCCTMRI	VCC Timer – This bit is set when the VCCTMR times out. This bit is cleared when the SCInt register is read.
SCInt.4	RXDAV	Rx Data Available – Data was received from the smart card because the Rx FIFO is not empty. In bypass mode, this interrupt is generated on a falling edge of the smart card I/O line. After receiving this interrupt in bypass mode, firmware should disable it until the firmware has received the entire byte and is waiting for the next start delimiter. This bit is cleared when there is no RX data available in the RX FIFO.
SCInt.3	TXEVNT	TX Event – Set whenever the TXEMTY or TXFULL bits are set in the SRXCtl SFR. This bit is cleared when the STXCtl register is read.
SCInt.2	TXSENT	TX Sent – Set whenever the ISO UART has successfully transmitted a byte to the smart card. Also set when a CRC/LRC byte is sent in T=1 mode. Will not be set in T=0 when a break is detected at the end of a byte (when break detection is enabled). This bit is cleared when the SCInt register is read.
SCInt.1	TXERR	TX Error – An error was detected during the transmission of data to the smart card as indicated by either BREAKD or TXUNDR bit being set in the STXCtl SFR. Additional information can be found in that register description. This bit is cleared when the STXCtl register is read.
SCInt.0	RXERR	RX Error – An error was detected during the reception of data from the smart card. Additional information can be found in the SRXCtl register. This interrupt will be asserted for RXOVRR, or RX Parity error events. This bit is cleared when the SRXCtl register is read.

Card Status/Control Register (CRDCtl): 0xFE05 ← 0x00

This register is used to configure the card detect pin (DETCARD) and monitor card detect status. This register must be written to properly configure Debounce, Detect_Polarity (= 0 or = 1), and the pull-up/down enable before setting CDETEN. The card detect logic is functional even without smart card logic clock. When the PWRDN bit is set = 1, no debounce is provided but card presence is operable.

Table 84: The CRDCtl Register

MSB LSB

DEBOUN	CDETEN	–	–	DETPOL	PUENB	PDEN	CARDIN

Bit	Symbol	Function
CRDCtl.7	DEBOUN	Debounce – When set = 1, this will enable hardware de-bounce of the card detect pin. The de-bounce function shall wait for 64ms of stable card detect assertion before setting the CARDIN bit. This counter/timer uses the keypad clock as a source of 1kHz signal. De-assertion of the CARDIN bit is immediate upon de-assertion of the card detect pin(s).
CRDCtl.6	CDETEN	Card Detect Enable – When set = 1, activates card detection input. Default upon power-on reset is 0.
CRDCtl.5	–	
CRDCtl.4	–	
CRDCtl.3	DETPOL	Detect Polarity – When set = 1, the DETCARD pin shall interpret a logic 1 as card present.
CRDCtl.2	PUENB	Enable pull-up current on DETCARD pin (active low).
CRDCtl.1	PDEN	Enable pull-down current on DETCARD pin.
CRDCtl.0	CARDIN	Card Inserted – (Read only). 1 = card inserted, 0 = card not inserted. A change in the value of this bit is a "card event." A read of this bit indicates whether smart card is inserted or not inserted in conjunction with the DETPOL setting.

TX Control/Status Register (STXCtl): 0xFE06 🡐 0x00

This register is used to control transmission of data to the smart card. Some control and some status bits are in this register.

Table 85: The STXCtl Register

MSB							LSB
I2CMODE	–	TXFULL	TXEMTY	TXUNDR	LASTTX	TX/RXB	BREAKD

Bit	Symbol	Function
STXCtl.7	I2CMODE	I2C Mode – When in sync mode and this bit is set, and when the RLen count value = max or 0, the source of the smart card data for IO pin (or SIO pin) will be connected to the IO bit in SCCtl (or SCECtl) register rather than the TX FIFO. See the description for the Protocol Mode Register for more detail.
STXCtl.6	–	
STXCtl.5	TXFULL	TX FIFO is full. Additional writes may corrupt the contents of the FIFO. This bit it will remain set as long as the TX FIFO is full. Generates a TX_Event interrupt upon going full.
STXCtl.4	TXEMTY	1 = TX FIFO is empty, 0 = TX FIFO is not empty. If there is data in the TX FIFO, the circuit will transmit it to the smart card if in transmit mode. In T=1 mode, if the LASTTX bit is set and the hardware is configured to transmit the CRC/LRC, the TXEMTY will not be set until the CRC/LRC is transmitted. In T=0, if the LASTTX bit is set, TXEMTY will be set after the last word has been successfully transmitted to the smart card. Generates a TXEVNT interrupt upon going empty.
STXCtl.3	TXUNDR	TX Underrrun – (Read only) Asserted when a transmit under-run condition has occurred. An under-run condition is defined as an empty TX FIFO when the last data word has been successfully transmitted to the smart card and the LASTTX bit was not set. No special processing is performed by the hardware if this condition occurs. Cleared when read by firmware. This bit generates a TXERR interrupt.
STXCtl.2	LASTTX	Last TX Byte – Set by firmware (in both T=0 and T=1) when the last byte in the current message has been written into the transmit FIFO. In T=1 mode, the CRC/LRC will be appended to the message. Should be set after the last byte has been written into the transmit FIFO. Should be cleared by firmware before writing first byte of next message into the transmit FIFO. Used in T=0 to determine when to set TXEMTY.
STXCtl.1	TX/RXB	1 = Transmit mode, 0 = Receive mode. Configures the hardware to be receiving from or transmitting to the smart card. Determines which counters should be enabled. This bit should be set to receive mode prior to switching to another interface. Setting and resetting this bit shall initialize the CRC logic. If LASTTX is set, this bit can be reset to RX mode and UART logic will automatically change mode to RX when TX operation is completed (TX_Empty =1).
STXCtl.0	BREAKD	Break Detected – (Read only) 1 = A break has been detected on the I/O line indicating that the smart card detected a parity error. Cleared when read. This bit generates a TXERR interrupt.

Block Guard Time Register (BGT): 0xFE16 ← 0x10

This register contains the Extra Guard Time Value (EGT) most-significant bit. The Extra Guard Time indicates the minimum time between the leading edges of the start bit of consecutive characters. The delay is depends on the T=0/T=1 mode. Used in transmit mode. This register also contains the Block Guard Time (BGT) value. Block Guard Time is the minimum time between the leading edge of the start bit of the last character received and the leading edge of the start bit of the first character transmitted. This should not be set less than the character length. The transmission of the first character will be held off until BGT has elapsed regardless of the TX data and TX/RX control bit timing.

Table 102: The BGT Register

MSB LSB

EGT.8	–	–	BGT.4	BGT.3	BGT.1	BGT.2	BGT.0

Bit	Symbol	Function
BGT.7	EGT.8	Most-significant bit for 9-bit EGT timer. See EGT below.
BGT.6	–	
BGT.5	–	
BGT.4	BGT.4	Time in ETUs between the start bit of the last received character to start bit of the first character transmitted to the smart card. Default value is 22.
BGT.3	BGT.3	
BGT.2	BGT.2	
BGT.1	BGT.1	
BGT.0	BGT.0	

Extra Guard Time Register (EGT): 0xFE17 ← 0x0C

This register contains the Extra Guard Time Value (EGT) least-significant byte. The Extra Guard Time indicates the minimum time between the leading edges of the start bit of consecutive characters. The delay depends on the T=0/T=1 mode. Used in transmit mode.

Table 103: The EGT Register

MSB LSB

EGT.7	EGT.6	EGT.5	EGT.4	EGT.3	EGT.1	EGT.2	EGT.0

Bit	Function
EGT.7	Time in ETUs between start bits of consecutive characters. In T=0 mode, the minimum is 1. In T=0, the leading edge of the next start bit may be delayed if there is a break detected from the smart card. Default value is 12. In T=0 mode, regardless of the value loaded, the minimum value is 12, and for T=1 mode, the minimum value is 11.
EGT.6	
EGT.5	
EGT.4	
EGT.3	
EGT.2	
EGT.1	
EGT.0	

Block Wait Time Registers (BWTB0): 0xFE1B ← 0x00, (BWTB1): 0xFE1A ← 0x00, (BWTB2): 0xFE19 ← 0x00, (BWTB3): 0xFE18 ← 0x00

Table 104: The BWTB0 Register

MSB LSB

BWT.7	BWT.6	BWT.5	BWT.4	BWT.3	BWT.1	BWT.2	BWT.0

Table 105: The BWTB1 Register

MSB LSB

BWT.15	BWT.14	BWT.13	BWT.12	BWT.11	BWT.10	BWT.9	BWT.8

Table 106: The BWTB2 Register

MSB LSB

BWT.23	BWT.22	BWT.21	BWT.20	BWT.19	BWT.18	BWT.17	BWT.16

Table 107: The BWTB3 Register

MSB LSB

–	–	–	–	BWT.27	BWT.26	BWT.25	BWT.24

These registers (BWTB0, BWTB1, BWTB2, BWTB3) are used to set the Block Waiting Time(27:0) (BWT). All of these parameters define the maximum time the 73S1217F will have to wait for a character from the smart card. These registers serve a dual purpose. When T=1, these registers are used to set up the block wait time. The block wait time defines the time in ETUs between the beginning of the last character sent to smart card and the start bit of the first character received from smart card. It can be used to detect an unresponsive card and should be loaded by firmware prior to writing the last TX byte. When T = 0, these registers are used to set up the work wait time. The work wait time is defined as the time between the leading edge of two consecutive characters being sent to or from the card. If a timeout occurs, an interrupt is generated to the firmware. The firmware can then take appropriate action. A Wait Time Extension (WTX) is supported with the 28-bit BWT.

Character Wait Time Registers (CWTB0): 0xFE1D ← 0x00, (CWTB1): 0xFE1C ← 0x00

Table 108: The CWTB0 Register

MSB LSB

CWT.7	CWT.6	CWT.5	CWT.4	CWT.3	CWT.1	CWT.2	CWT.0

Table 109: The CWTB1 Register

MSB LSB

CWT.15	CWT.14	CWT.13	CWT.12	CWT.11	CWT.10	CWT.9	CWT.8

These registers (CWTB0, CWTB1) are used to hold the Character Wait Time(15:0) (CWT) or Initial Waiting Time(15:0) (IWT) depending on the situation. Both the IWT and the CWT measure the time in ETUs between the leading edge of the start of the current character received from the smart card and the leading edge of the start of the next character received from the smart card. The only difference is the mode in which the card is operating. When T=1 these registers are used to configure the CWT and these registers configure the IWT when the ATR is being received. These registers should be loaded prior to receiving characters from the smart card. Firmware must manage which time is stored in the register. If a timeout occurs, an interrupt is generated to the firmware. The firmware can then take appropriate action.

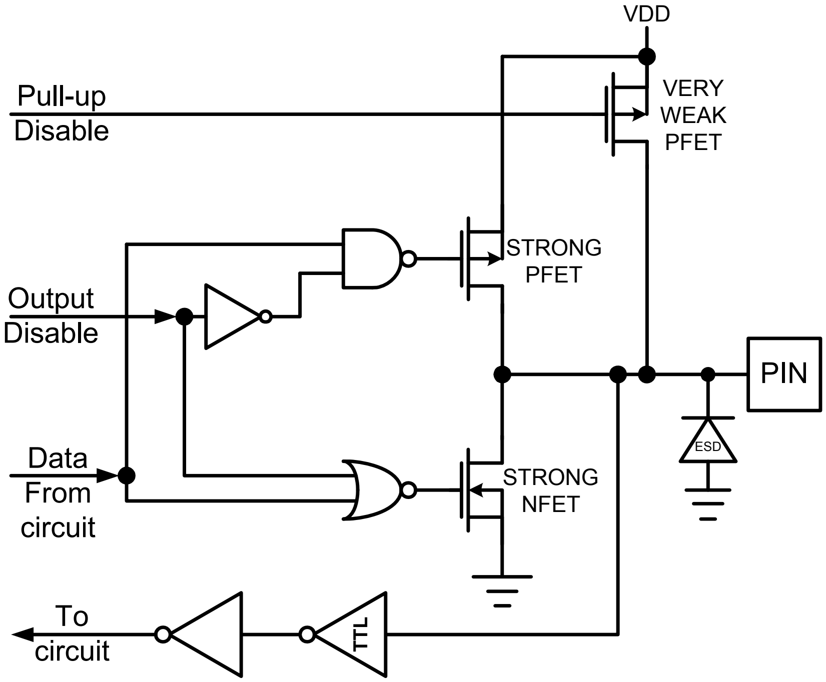


Figure 33: Digital I/O with Pull Up Circuit

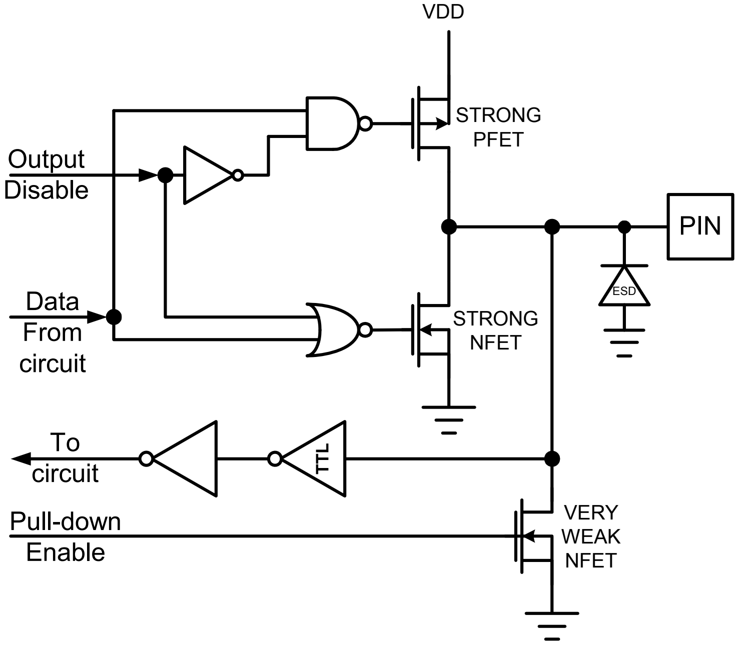


Figure 34: Digital I/O with Pull Down Circuit

4.1 Package Pin Designation (68-Pin QFN)

CAUTION: Use handling procedures necessary for a static sensitive component

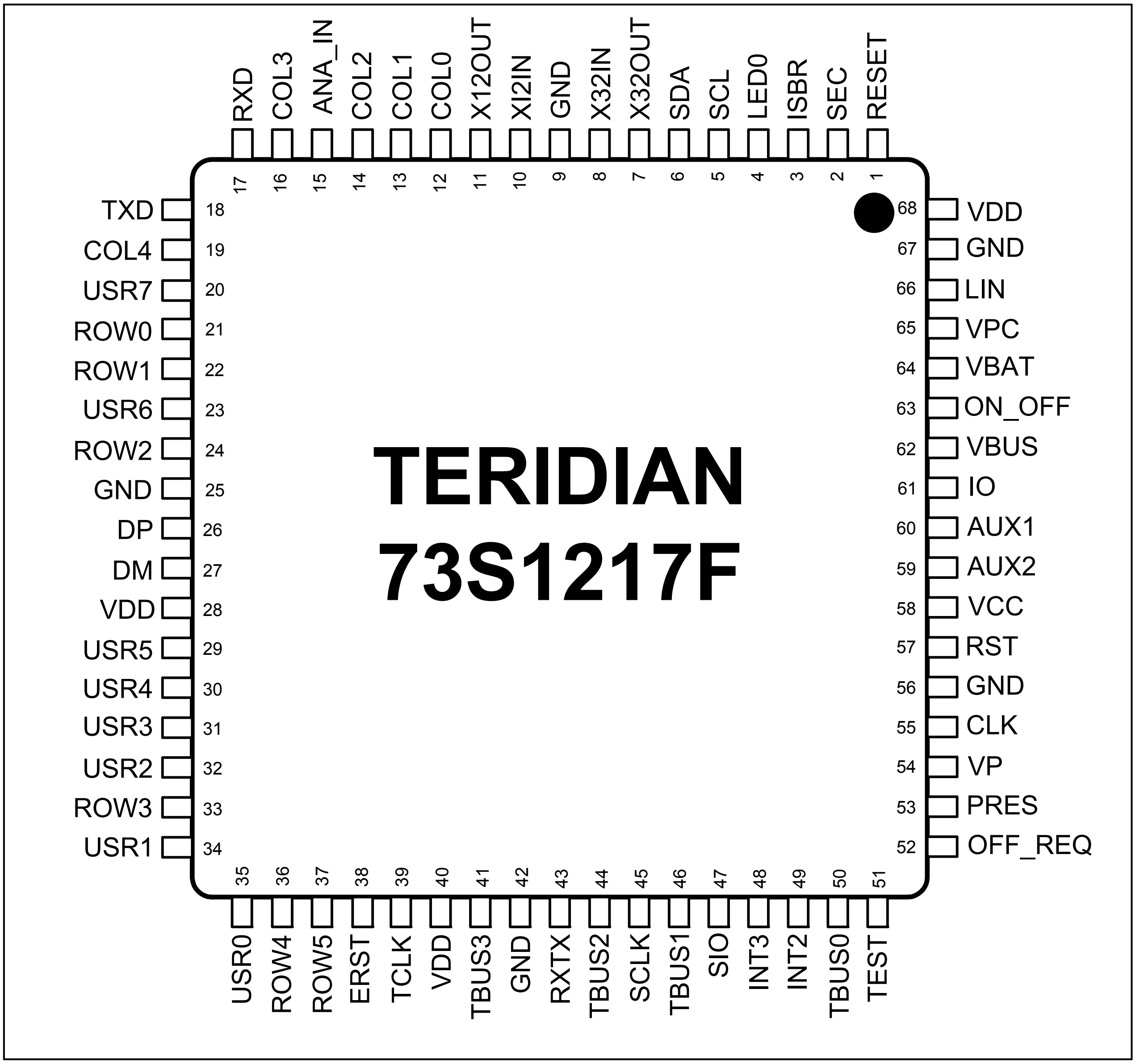


Figure 47: 73S1217F Pinout

4.2 Packaging Information

68-Pin QFN Package Outline

Notes: 6.3mm x 6.3mm exposed pad area must remain UNCONNECTED (clear of PCB traces or vias). Controlling dimensions are in mm.

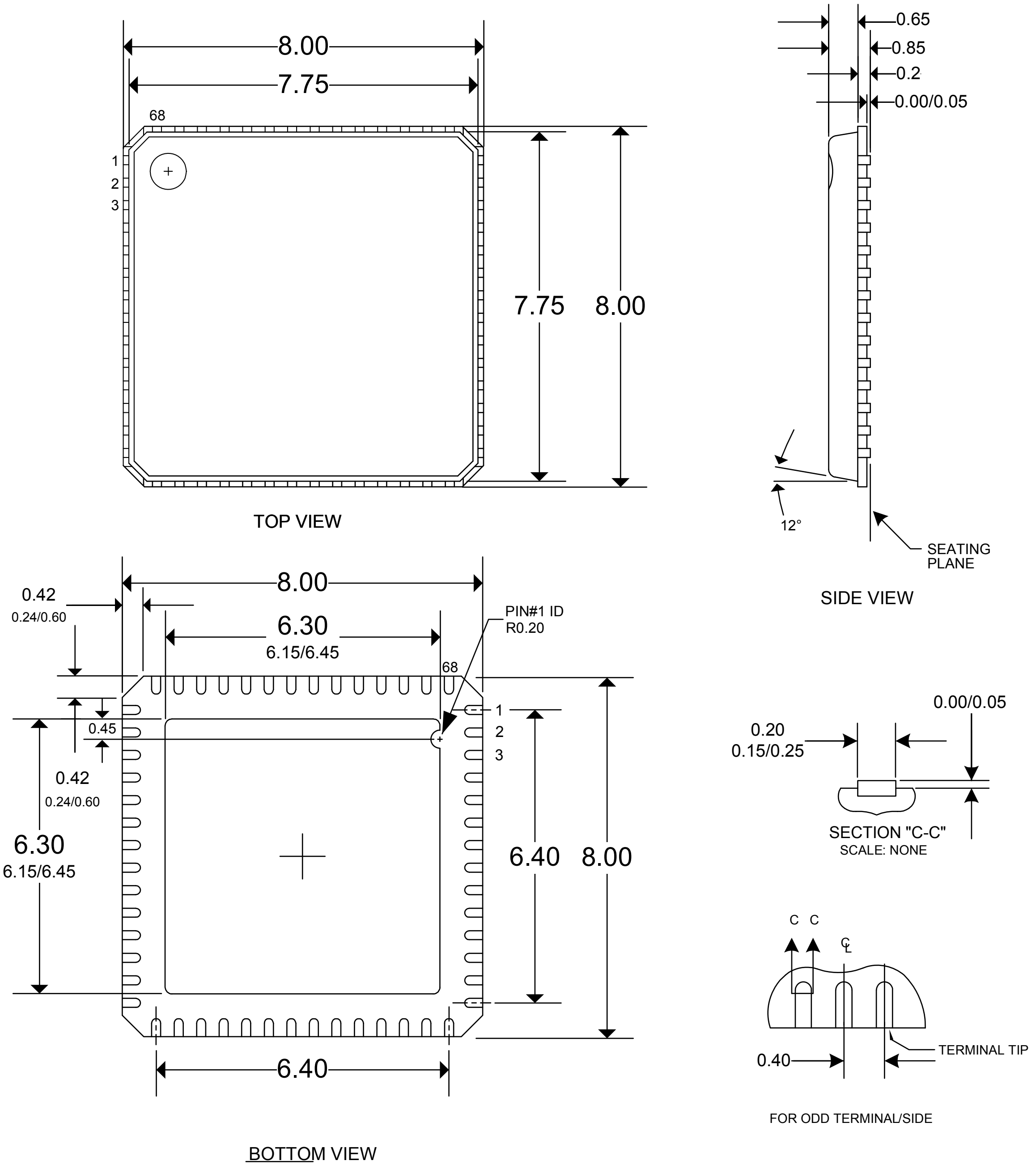


Figure 48: 73S1217F 68 QFN Mechanical Drawing