Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of "Quality Parts,Customers Priority,Honest Operation,and Considerate Service",our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

MAXIM
Teridian Smart Grid Solutions

# 71M6521DE/DH/FE
Energy Meter ICs

DATA SHEET

GENERAL DESCRIPTION

The Teridian™ 71M6521DE/DH/FE energy meter ICs are highly integrated systems-on-a-chip (SoCs) with an MPU core, RTC, flash, and LCD driver. The Single Converter Technology® with a 22-bit delta-sigma ADC, four analog inputs, digital temperature compensation, precision voltage reference, battery voltage monitor, and 32-bit computation engine (CE) supports a wide range of residential metering applications with very few low-cost external components. A 32kHz crystal time base for the entire system and internal battery-backup support for RAM and RTC further reduce system cost. The ICs support 2-wire, 3-wire, and 4-wire single-phase and dual-phase residential metering along with tamper-detection mechanisms.

Maximum design flexibility is provided by multiple UARTs, I²C, MICROWIRE®, up to 18 DIO pins, and in-system programmable flash memory, which can be updated with data or application code in operation.

A complete array of ICE and development tools, programming libraries, and reference designs enable rapid development and certification of TOU, AMR, and prepay meters that comply with worldwide electricity metering standards.
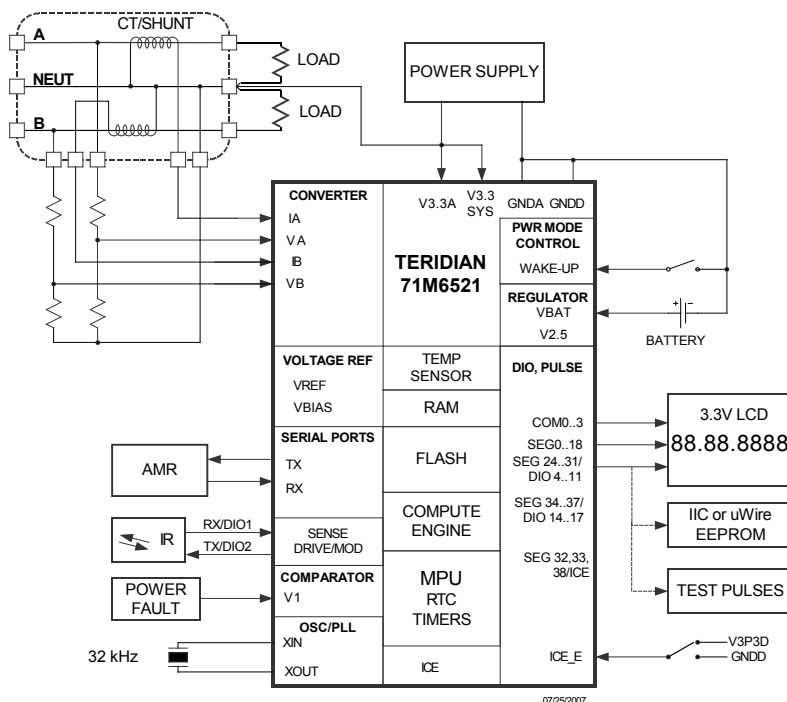
Teridian is a trademark and Single Converter Technology is a registered trademark of Maxim Integrated Products Inc.
MICROWIRE is a registered trademark of National Semiconductor Corp.

FEATURES

- Up to 0.1% Wh Accuracy Over 2000:1 Current Range
- Exceeds IEC 62053/ANSI C 12.20 Standards
- Voltage Reference < 40 ppm/°C (< 20ppm/°C for 71M6521DH)
- Four Sensor Inputs—V_{DD} Referenced
- Low-Jitter Wh and VARh Pulse Test Outputs (10kHz max)
- Pulse Count for Pulse Outputs
- Four-Quadrant Metering
- Tamper Detection
    Neutral Current with CT or Shunt
- Line Frequency Count for RTC
- Digital Temperature Compensation
- Sag Detection for Phase A and B
- Independent 32-Bit Compute Engine
- 46-64Hz Line Frequency Range with Same Calibration
- Phase Compensation (±7°)
- Battery backup for RTC and battery monitor
- Three Battery Modes with Wake-Up on Pushbutton or Timer:
    Brownout Mode (48µA)
    LCD Mode (5.7µA)
    Sleep Mode (2.9µA)
- Energy Display on Main Power Failure
- Wake-Up with Pushbutton
- 22-Bit Delta-Sigma ADC
- 8-Bit MPU (80515), 1 Clock Cycle per Instruction with Integrated ICE for MPU Debug
- RTC with Temperature Compensation
- Auto-Calibration
- Hardware Watchdog Timer, Power-Fail Monitor
- LCD Driver (Up to 152 Pixels)
- Up to 18 General-Purpose I/O Pins
- 32kHz Time Base
- 16KB (6521DE/DH) or 32KB (6521FE) Flash with Security
- 2KB MPU XRAM
- Two UARTs for IR and AMR
- Digital I/O Pins Compatible with 5V Inputs
- 64-Pin LQFP or 68-Pin QFN Package
- Lead(Pb)-Free Packages

Table of Contents

List of Figures

List of Tables

Figure 1: IC Functional Block Diagram

HARDWARE DESCRIPTION

Hardware Overview

The Teridian 71M6521DE/DH/FE single-chip energy meter integrates all primary functional blocks required to implement a solid-state electricity meter. Included on chip are an analog front end (AFE), an independent digital computation engine (CE), an 8051-compatible microprocessor (MPU) which executes one instruction per clock cycle (80515), a voltage reference, a temperature sensor, LCD drivers, RAM, flash memory, a real time clock (RTC), and a variety of I/O pins. Various current sensor technologies are supported including Current Transformers (CT), and Resistive Shunts.

In a typical application, the 32-bit compute engine (CE) of the 71M6521DE/DH/FE sequentially processes the samples from the voltage inputs on pins IA, VA, IB, VB and performs calculations to measure active energy (Wh), reactive energy (VARh), A^2h, and V^2h for four-quadrant metering. These measurements are then accessed by the MPU, processed further and output using the peripheral devices available to the MPU.

In addition to advanced measurement functions, the real time clock function allows the 71M6521DE/DH/FE to record time of use (TOU) metering information for multi-rate applications and to time-stamp tamper events. Measurements can be displayed on 3.3V LCD commonly used in low temperature environments. Flexible mapping of LCD display segments will facilitate integration of existing custom LCD. Design trade-off between number of LCD segments vs. DIO pins can be implemented in software to accommodate various requirements.

In addition to the temperature-trimmed ultra-precision voltage reference, the on-chip digital temperature compensation mechanism includes a temperature sensor and associated controls for correction of unwanted temperature effects on measurement and RTC accuracy, e.g. to meet the requirements of ANSI and IEC standards. Temperature dependent external components such as crystal oscillator, current transformers (CTs), and their corresponding signal conditioning circuits can be characterized and their correction factors can be programmed to produce electricity meters with exceptional accuracy over the industrial temperature range.

The 71M6521DH is trimmed at +85°C in addition to the trim at room temperature, which provides a set of individualized trim fuse values that enable temperature compensation with accuracy better than ±20 PPM/°C.

One of the two internal UARTs is adapted to support an Infrared LED with internal drive and sense configuration, and can also function as a standard UART. The optical output can be modulated at 38 kHz. This flexibility makes it possible to implement AMR meters with an IR interface. A block diagram of the IC is shown in Figure 1. A detailed description of various functional blocks follows.

Analog Front End (AFE)

The AFE of the 71M6521DE/DH/FE is composed of an input multiplexer, a delta-sigma A/D converter and a voltage reference.

Input Multiplexer

The input multiplexer supports up to four input signals that are applied to pins IA, VA, IB and VB of the device. Additionally, using the alternate mux selection, it has the ability to select temperature and the battery voltage. The multiplexer can be operated in two modes:

- During a normal multiplexer cycle, the signals from the IA, IB, VA, and VB pins are selected.
- During the alternate multiplexer cycle, the temperature signal (TEMP) and the battery monitor are selected, along with the signal sources shown in Table 1. To prevent unnecessary drainage on the battery, the battery monitor is enabled only with the *BME* bit (0x2020[6]) in the I/O RAM.

The alternate mux cycles are usually performed infrequently (e.g. every second) by the MPU. In order to prevent disruption of the voltage tracking PLL and voltage allpass networks, VA is not replaced in the ALT mux selections. Table 1 details the regular and alternative MUX sequences. Missing samples due to an ALT multiplexer sequence are filled in by the CE.

| | Regular MUX Sequence | | | | ALT MUX Sequence | | | |
| | Mux State | | | | Mux State | | | |
EQU	0	1	2	3	0	1	2	3
0, 1, 2	IA	VA	IB	VB	TEMP	VA	VBAT	VB

Table 1: Inputs Selected in Regular and Alternate Multiplexer Cycles

In a typical application, IA and IB are connected to current transformers that sense the current on each phase of the line voltage. VA and VB are typically connected to voltage sensors through resistor dividers.

The multiplexer control circuit handles the setting of the multiplexer. The function of the control circuit is governed by the I/O RAM registers *MUX_ALT, MUX_DIV* and *EQU. MUX_DIV* controls the number of samples per cycle. It can request 2, 3, or 4 multiplexer states per cycle. Multiplexer states above 4 are reserved and must not be used. The multiplexer always starts at the beginning of its list and proceeds until *MUX_DIV* states have been converted.

The *MUX_ALT* bit requests an alternative multiplexer frame. The bit may be asserted on any MPU cycle and may be subsequently de-asserted on any cycle including the next one. A rising edge on *MUX_ALT* will cause the multiplexer control circuit to wait until the next multiplexer cycle and implement a single alternate cycle.

The multiplexer control circuit also controls the FIR filter initiation and the chopping of the ADC reference voltage, VREF. The multiplexer control circuit is clocked by CK32, the 32768Hz clock from the PLL block, and launches with each new pass of the CE program.

A/D Converter (ADC)

A single delta-sigma A/D converter digitizes the voltage and current inputs to the 71M6521DE/DH/FE. The resolution of the ADC is programmable using the *FIR_LEN* register as shown in the I/O RAM section. ADC resolution can be selected to be 21 bits (*FIR_LEN*=0), or 22 bits (*FIR_LEN*=1). Conversion time is two cycles of CK32 with *FIR_LEN* = 0 and three cycles with *FIR_LEN* = 1.

In order to provide the maximum resolution, the ADC should be operated with *FIR_LEN* = 1. Accuracy and timing specifications in this data sheet are based on *FIR_LEN* = 1.

Initiation of each ADC conversion is controlled by the multiplexer control circuit as described previously. At the end of each ADC conversion, the FIR filter output data is stored into the CE DRAM location determined by the multiplexer selection.

FIR Filter

The finite impulse response filter is an integral part of the ADC and it is optimized for use with the multiplexer. The purpose of the FIR filter is to decimate the ADC output to the desired resolution. At the end of each ADC conversion, the output data is stored into the fixed CE DRAM location determined by the multiplexer selection. FIR data is stored LSB justified, but shifted left by nine bits.

Voltage References

The device includes an on-chip precision bandgap voltage reference that incorporates auto-zero techniques. The reference is trimmed to minimize errors caused by component mismatch and drift. The result is a voltage output with a predictable temperature coefficient.

The amplifier within the reference is chopper stabilized, i.e. the polarity can be switched by the MPU using the I/O RAM register *CHOP_E* (0x2002[5:4]). The two bits in the *CHOP_E* register enable the MPU to operate the chopper circuit in regular or inverted operation, or in "toggling" mode. When the chopper circuit is toggled in between multiplexer cycles, DC offsets on the measured signals will automatically be averaged out.

The general topology of a chopped amplifier is given in Figure 2.
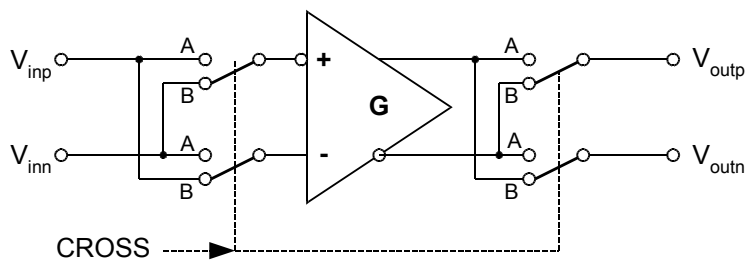
Figure 2: General Topology of a Chopped Amplifier

It is assumed that an offset voltage Voff appears at the positive amplifier input. With all switches, as controlled by CROSS in the "A" position, the output voltage is:

Voutp – Voutn = G (Vinp + Voff – Vinn) = G (Vinp – Vinn) + G Voff

With all switches set to the "B" position by applying the inverted CROSS signal, the output voltage is:

Voutn – Voutp = G (Vinn – Vinp + Voff) = G (Vinn – Vinp) + G Voff, or

Voutp – Voutn = G (Vinp – Vinn) - G Voff

Thus, when CROSS is toggled, e.g. after each multiplexer cycle, the offset will alternately appear on the output as positive and negative, which results in the offset effectively being eliminated, regardless of its polarity or magnitude.

When CROSS is high, the hookup of the amplifier input devices is reversed. This preserves the overall polarity of that amplifier gain; it inverts its input offset. By alternately reversing the connection, the amplifier's offset is averaged to zero. This removes the most significant long-term drift mechanism in the voltage reference. The *CHOP_E* bits control the behavior of CROSS. The CROSS signal will reverse the amplifier connection in the voltage reference in order to negate the effects of its offset. On the first CK32 rising edge after the last mux state of its sequence, the mux will wait one additional CK32 cycle before beginning a new frame. At the beginning of this cycle, the value of CROSS will be updated according to the *CHOP_E* bits. The extra CK32 cycle allows time for the chopped VREF to settle. During this cycle, MUXSYNC is held high. The leading edge of muxsync initiates a pass through the CE program sequence. The beginning of the sequence is the serial readout of the 4 RTM words.

CHOP_E has 3 states: positive, reverse, and chop. In the 'positive' state, CROSS is held low. In the 'reverse' state, CROSS is held high. In the 'chop' state, CROSS is toggled near the end of each Mux Frame, as described above. It is desirable that CROSS take on alternate values at the beginning of each Mux cycle. For this reason, if 'chop' state is selected, CROSS will not toggle at the end of the last Mux cycle in a SUM cycle.

The internal bias voltage VBIAS (typically 1.6 V) is used by the ADC when measuring the temperature and battery monitor signals.

Temperature Sensor

The 71M6521DE/DH/FE includes an on-chip temperature sensor implemented as a bandgap reference. It is used to determine the die temperature The MPU may request an alternate multiplexer cycle containing the temperature sensor output by asserting *MUX_ALT*.

The primary use of the temperature data is to determine the magnitude of compensation required to offset the thermal drift in the system (see section titled "Temperature Compensation").

Battery Monitor

The battery voltage is measured by the ADC during alternative multiplexer frames if the *BME* (Battery Measure Enable) bit in the I/O RAM is set. While *BME* is set, an on-chip 45kΩ load resistor is applied to the battery, and a scaled fraction of the battery voltage is applied to the ADC input. After each alternative MUX frame, the result of the ADC conversion is available at CE DRAM address 07. *BME* is ignored and assumed zero when system power is not available (V1 < VBIAS). See the Battery Monitor section of the Electrical Specifications for details regarding the ADC LSB size and the conversion accuracy.

Functional Description

The AFE functions as a data acquisition system, controlled by the MPU. The main signals (IA, VA, IB, VB) are sampled and the ADC counts obtained are stored in CE DRAM where they can be accessed by the CE and, if necessary, by the MPU. Alternate multiplexer cycles are initiated less frequently by the MPU to gather access to the slow temperature and battery signals.
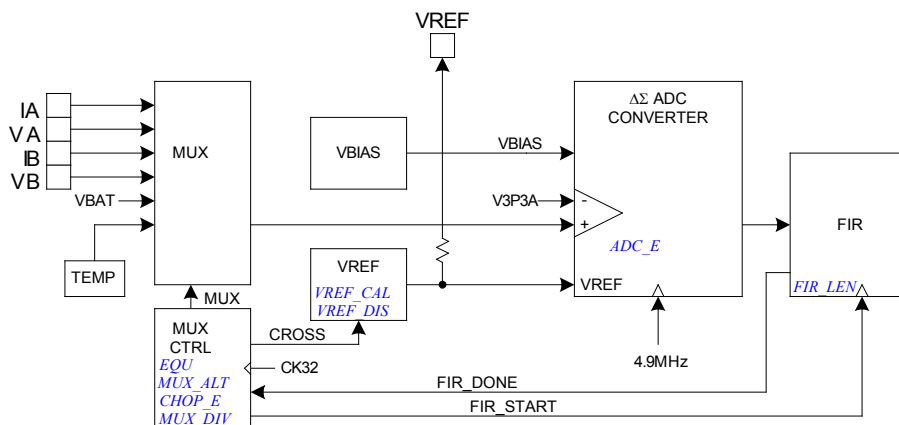
Figure 3: AFE Block Diagram

Digital Computation Engine (CE)

The CE, a dedicated 32-bit signal processor, performs the precision computations necessary to accurately measure energy. The CE calculations and processes include:

- Multiplication of each current sample with its associated voltage sample to obtain the energy per sample (when multiplied with the constant sample time).

- Frequency-insensitive delay cancellation on all six channels (to compensate for the delay between samples caused by the multiplexing scheme).

- 90° phase shifter (for VAR calculations).

- Pulse generation.

- Monitoring of the input signal frequency (for frequency and phase information).

- Monitoring of the input signal amplitude (for sag detection).

- Scaling of the processed samples based on calibration coefficients.

The CE program resides in flash memory. Common access to flash memory by CE and MPU is controlled by a memory share circuit. Each CE instruction word is two bytes long. Allocated flash space for the CE program cannot exceed 1024 words (2KB). The CE program counter begins a pass through the CE code each time multiplexer state 0 begins. The code pass ends when a HALT instruction is executed. For proper operation, the code pass must be completed before the multiplexer cycle ends (see System Timing Summary in the Functional Description Section).

The CE program must begin on a 1Kbyte boundary of the flash address. The I/O RAM register *CE_LCTN[4:0]* defines which 1KB boundary contains the CE code. Thus, the first CE instruction is located at 1024*CE_LCTN[4:0].

The CE DRAM can be accessed by the FIR filter block, the RTM circuit, the CE, and the MPU. Assigned time slots are reserved for FIR, RTM, and MPU, respectively, to prevent bus contention for CE DRAM data access. Holding re-gisters are used to convert 8-bit wide MPU data to/from 32-bit wide CE DRAM data, and wait states are inserted as needed, depending on the frequency of CKMPU.

The CE DRAM contains 128 32-bit words. The MPU can read and write the CE DRAM as the primary means of data communication between the two processors.

Table 2 shows the CE DRAM addresses allocated to analog inputs from the AFE.

ADDRESS (HEX)	NAME	DESCRIPTION
00	IA	Phase A current
01	VA	Phase A voltage
02	IB	Phase B current
03	VB	Phase B voltage
04	-	Not used
05	-	Not used
06	TEMP	Temperature
07	VBAT	Battery Voltage

Table 2: CE DRAM Locations for ADC Results

The CE of the 71M6521DE/DH/FE is aided by support hardware that facilitates implementation of equations, pulse counters, and accumulators. This support hardware is controlled through I/O RAM locations *EQU* (equation assist), *DIO_PV* and *DIO_PW* (pulse count assist), and *PRE_SAMPS* and *SUM_CYCLES* (accumulation assist). *PRE_SAMPS* and *SUM_CYCLES* support a dual level accumulation scheme where the first accumulator accumulates results from *PRE_SAMPS* samples and the second accumulator accumulates up to *SUM_CYCLES* of the first accumulator results. The integration time for each energy output is *PRE_SAMPS * SUM_CYCLES*/2520.6 (with *MUX_DIV* = 1). CE hardware issues the XFER_BUSY interrupt when the accumulation is complete.

Meter Equations

Compute Engine (CE) firmware and hardware for residential meter configurations implement the equations listed in Table 3. The register *EQU* (located in the I/O RAM) specifies the equation to be used based on the number of phases used for metering.

EQU	Description	Watt & VAR Formula	
		Element 0	Element 1
0	1 element, 2W 1φ with neutral current sense and tamper detection (VA connected to VB)	VA IA	VA IB
1	1 element, 3W 1φ	VA(IA-IB)/2	N/A
2	2 element, 4W 2φ	VA IA	VB IB

Table 3: Meter Equations.

Real-Time Monitor

The CE contains a Real-Time Monitor (RTM), which can be programmed through the UART to monitor four selectable CE DRAM locations at full sample rate. The four monitored locations are serially output to the TMUXOUT pin via the digital output multiplexer at the beginning of each CE code pass. The RTM can be enabled and disabled with *RTM_EN*. The RTM output is clocked by CKTEST. Each RTM word is clocked out in 35 cycles and contains a leading flag bit. See the Functional Description section for the RTM output format. RTM is low when not in use.

Pulse Generator

The chip contains two pulse generators that create low-jitter pulses at a rate set by either CE or MPU. The function is distinguished by *EXT_PULSE* (a CE input variable in CE DRAM):

- If *EXT_PULSE* = 1, *APULSEW*WRATE* and *APULSER*WRATE* control the pulse rate (external pulse generation)

- If *EXT_PULSE* is 0, *APULSEW* is replaced with *WSUM_X* and *APULSER* is replaced with *VARSUM_X* (internal pulse generation).

The I/O RAM bits *DIO_PV* and *DIO_PW,* as described in the Digital I/O section, can be programmed to route WPULSE to the output pin DIO6 and VARPULSE to the output pin DIO7. Pulses can also be output on OPT_TX (see *OPT_TXE[1:0]* for details).

During each CE code pass, the hardware stores exported sign bits in an 8-bit FIFO and outputs them at a specified interval. This permits the CE code to calculate all of the pulse generator outputs at the beginning of its code pass and to rely on hardware to spread them over the MUX frame. The FIFO is reset at the beginning of each MUX frame. *PLS_INTERVAL* controls the delay to the first pulse update and the interval between subsequent updates. Its LSB is four CK_FIR cycles, or 4 * 203ns. If *PLS_INTERVAL* is zero, the FIFO is deactivated and the pulse outputs are updated immediately. Thus, $N_{INTERVAL}$ is *4*PLS_INTERVAL.*

For use with the supplied standard Teridian CE code, *PLS_INTERVAL* is set to a fixed value of 81. *PLS_INTERVAL* is specified so that all of the pulse updates are output <u>before</u> the MUX frame completes.

On-chip hardware provides a maximum pulse width feature: *PLS_MAXWIDTH[7:0]* selects a maximum negative pulse width to be 'Nmax' updates per multiplexer cycle according to the formula: Nmax = (2**PLS_MAXWIDTH*+1). If *PLS_MAXWIDTH* = 255, no width checking is performed.

Given that *PLS_INTERVAL* = 81, the maximum pulse width is determined by:

Maximum Pulse Width = (2 * *PLS_MAXWIDTH* +1) * 81*4*203ns = 65.9µs + *PLS_MAXWIDTH* * 131.5µs

If the pulse period corresponding to the pulse rate exceeds the desired pulse width, a square wave with 50% duty-cycle is generated.

The CE pulse output polarity is programmable to be either positive or negative. Pulse polarity may be inverted with *PLS_INV*. When this bit is set, the pulses are active high, rather than the more usual active low.

CE Functional Overview

The ADC processes one sample per channel per multiplexer cycle. Figure 4 shows the timing of the samples taken during one multiplexer cycle.

The number of samples processed during one accumulation cycle is controlled by the I/O RAM registers *PRE_SAMPS* (0x2001[7:6]) and *SUM_CYCLES* (0x2001[5:0]). The integration time for each energy output is

PRE_SAMPS * *SUM_CYCLES* / 2520.6, where 2520.6 is the sample rate [Hz]

For example, *PRE_SAMPS* = 42 and *SUM_CYCLES* = 50 will establish 2100 samples per accumulation cycle. *PRE_SAMPS* = 100 and *SUM_CYCLES* = 21 will result in the exact same accumulation cycle of 2100 samples or 833ms. After an accumulation cycle is completed, the XFER_BUSY interrupt signals to the MPU that accumulated data are available.
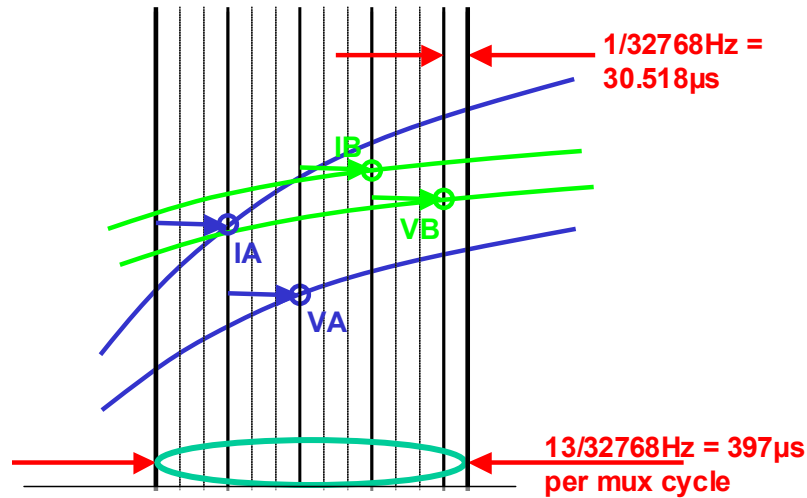
Figure 4: Samples from Multiplexer Cycle

The end of each multiplexer cycle is signaled to the MPU by the CE_BUSY interrupt. At the end of each multiplexer cycle, status information, such as sag data and the digitized input signal, is available to the MPU.
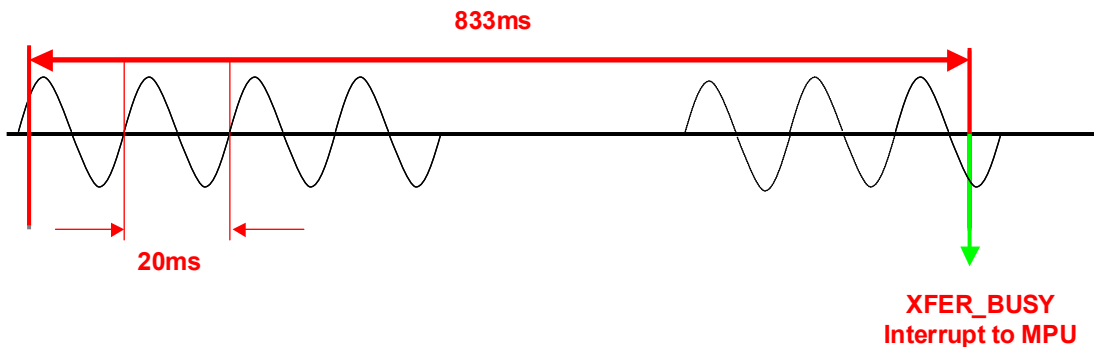
Figure 5: Accumulation Interval

Figure 5 shows the accumulation interval resulting from *PRE_SAMPS* = 42 and *SUM_CYCLES* = 50, consisting of 2100 samples of 397µs each, followed by the XFER_BUSY interrupt. The sampling in this example is applied to a 50Hz signal.

There is no correlation between the line signal frequency and the choice of *PRE_SAMPS* or *SUM_CYCLES (*even though when *SUM_CYCLES* = 42 one set of *SUM_CYCLES* happens to sample a period of 16.6ms). Furthermore, sampling does not have to start when the line voltage crosses the zero line, and the length of the accumulation interval need not be an integer multiple of the signal cycles.

It is important to note that the length of the accumulation interval, as determined by N_{ACC}, the product of *SUM_CYCLES* and *PRE_SAMPS,* is not an exact multiple of 1000ms. For example, if *SUM_CYCLES* = 60, and *PRE_SAMPS* = 00 (42), the resulting accumulation interval is:

$$\tau = \frac{N_{ACC}}{f_S} = \frac{60 \cdot 42}{\frac{32768Hz}{13}} = \frac{2520}{2520.62Hz} = 999.75ms$$

This means that accurate time measurements should be based on the RTC, not the accumulation interval.

80515 MPU Core

The 71M6521DE/DH/FE includes an 80515 MPU (8-bit, 8051-compatible) that processes most instructions in one clock cycle. Using a 5 MHz clock results in a processing throughput of 5 MIPS. The 80515 architecture eliminates redundant bus states and implements parallel execution of fetch and execution phases. Normally a machine cycle is aligned with a memory fetch, therefore, most of the 1-byte instructions are performed in a single cycle. This leads to an 8x performance (in average) improvement (in terms of MIPS) over the Intel 8051 device running at the same clock frequency.

Actual processor clocking speed can be adjusted to the total processing demand of the application (metering calculations, AMR management, memory management, LCD driver management and I/O management) using the I/O RAM register *MPU_DIV[2:0]*.

Typical measurement and metering functions based on the results provided by the internal 32-bit compute engine (CE) are available for the MPU as part of the Teridian standard library. A standard ANSI "C" 80515-application programming interface library is available to help reduce design cycle.

Memory Organization

The 80515 MPU core incorporates the Harvard architecture with separate code and data spaces.

Memory organization in the 80515 is similar to that of the industry standard 8051. There are three memory areas: Program memory (flash), external data memory (XRAM), physically consisting of XRAM, CE DRAM, and I/O RAM, and internal data memory (Internal RAM). Table 4 shows the memory map.

Address (hex)	Memory Technology	Memory Type	Typical Usage	Wait States (at 5MHz)	Memory Size (bytes)
0000-7FFF 0000-3FFF 0000-1FFF	Flash Memory	Non-volatile	MPU Program and non-volatile data	0	32K 16K 8K
on 1K boundary	Flash Memory	Non-volatile	CE program	0	2K
0000-07FF	Static RAM	Volatile	MPU data XRAM,	0	2K
1000-11FF	Static RAM	Volatile	CE data	6	512
2000-20FF	Static RAM	Volatile	Configuration RAM I/O RAM	0	256

Table 4: Memory Map

Internal and External Data Memory: Both internal and external data memory are physically located on the 71M6521DE/DH/FE IC. "External" data memory is only external to the 80515 MPU core.

Program Memory: The 80515 can theoretically address up to 64KB of program memory space from 0x0000 to 0xFFFF. Program memory is read when the MPU fetches instructions or performs a MOVC operation.

After reset, the MPU starts program execution from location 0x0000. The lower part of the program memory includes reset and interrupt vectors. The interrupt vectors are spaced at 8-byte intervals, starting from 0x0003.

External Data Memory: While the 80515 is capable of addressing up to 64KB of external data memory (0x0000 to 0xFFFF), **only the memory ranges shown in Table 4: Memory Map**

contain physical memory. The 80515 writes into external data memory when the MPU executes a MOVX @Ri,A or MOVX @DPTR,A instruction. The MPU reads external data memory by executing a MOVX A,@Ri or MOVX A,@DPTR instruction (SFR USR2 provides the upper 8 bytes for the MOVX A,@Ri instruction).

Clock Stretching: MOVX instructions can access fast or slow external RAM and external peripherals. The three low order bits of the *CKCON* register define the stretch memory cycles. Setting all the *CKCON* stretch bits to one allows access to very slow external RAM or external peripherals.

Table 5 shows how the signals of the External Memory Interface change when stretch values are set from 0 to 7. The widths of the signals are counted in MPU clock cycles. The post-reset state of the CKCON register, which is in bold in the table, performs the MOVX instructions with a stretch value equal to 1.

CKCON register			Stretch Value	Read signals width		Write signal width	
CKCON.2	*CKCON.1*	*CKCON.0*		memaddr	memrd	memaddr	memwr
0	0	0	0	1	1	2	1
0	**0**	**1**	**1**	**2**	**2**	**3**	**1**
0	1	0	2	3	3	4	2
0	1	1	3	4	4	5	3
1	0	0	4	5	5	6	4
1	0	1	5	6	6	7	5
1	1	0	6	7	7	8	6
1	1	1	7	8	8	9	7

Table 5: Stretch Memory Cycle Width

There are two types of instructions, differing in whether they provide an eight-bit or sixteen-bit indirect address to the external data RAM.

In the first type (MOVX A,@Ri), the contents of R0 or R1, in the current register bank, provide the eight lower-ordered bits of address. The eight high-ordered bits of address are specified with the *USR2* SFR. This method allows the user paged access (256 pages of 256 bytes each) to all ranges of the external data RAM. In the second type of MOVX instruction (MOVX A,@DPTR), the data pointer generates a sixteen-bit address. This form is faster and more efficient when accessing very large data arrays (up to 64 Kbytes), since no additional instructions are needed to set up the eight high ordered bits of address.

It is possible to mix the two MOVX types. This provides the user with four separate data pointers, two with direct access and two with paged access to the entire 64KB of external memory range.

Dual Data Pointer: The Dual Data Pointer accelerates the block moves of data. The standard *DPTR* is a 16-bit register that is used to address external memory or peripherals. In the 80515 core, the standard data pointer is called DPTR, the second data pointer is called *DPTR1*. The data pointer select bit chooses the active pointer. The data pointer select bit is located at the LSB of the *DPS* register (*DPS.0*). *DPTR* is selected when *DPS.0* = 0 and *DPTR1* is selected when *DPS.0* = 1.

The user switches between pointers by toggling the LSB of the DPS register. All data pointer-related instructions use the currently selected data pointer for any activity.

The second data pointer may not be supported by certain compilers.

Internal Data Memory: The Internal data memory provides 256 bytes (0x00 to 0xFF) of data memory. The internal data memory address is always 1 byte wide and can be accessed by either direct or indirect addressing. The Special Function Registers occupy the upper 128 bytes**. This SFR area is available only by direct addressing. Indirect addressing accesses the upper 128 bytes of Internal RAM.**

Internal Data Memory: The lower 128 bytes contain working registers and bit-addressable memory. The lower 32 bytes form four banks of eight registers (R0-R7). Two bits on the program memory status word (PSW) select which bank is in use. The next 16 bytes form a block of bit-addressable memory space at bit addressees 0x00-0x7F. All of the bytes in the lower 128 bytes are accessible through direct or indirect addressing. Table 6 shows the internal data memory map.

Address	Direct addressing	Indirect addressing
0xFF	Special Function Registers (SFRs)	**RAM**
0x80		
0x7F	Byte-addressable area	
0x30		
0x2F	Bit-addressable area	
0x20		
0x1F	**Register banks R0…R7**	
0x00		

Table 6: Internal Data Memory Map

Special Function Registers (SFRs)

A map of the Special Function Registers is shown in Table 7.

Hex\Bin	Bit-address-able	Byte-addressable							Bin/Hex
	X000	X001	X010	X011	X100	X101	X110	X111	
F8	*INTBITS*								FF
F0	*B*								F7
E8	*WDI*								EF
E0	*A*								E7
D8	*WDCON*								DF
D0	*PSW*								D7
C8	*T2CON*								CF
C0	*IRCON*								C7
B8	*IEN1*	*IP1*	*S0RELH*	*S1RELH*				*USR2*	BF
B0			***FLSHCTL***					***PGADR***	B7
A8	*IEN0*	*IP0*	*S0RELL*						AF
A0	***P2***	***DIR2***	***DIR0***						A7
98	*S0CON*	*S0BUF*	*IEN2*	*S1CON*	*S1BUF*	*S1RELL*	***EEDATA***	***EECTRL***	9F
90	***P1***	***DIR1***	*DPS*		***ERASE***				97
88	*TCON*	*TMOD*	*TL0*	*TL1*	*TH0*	*TH1*	*CKCON*		8F
80	***P0***	*SP*	*DPL*	*DPH*	*DPL1*	*DPH1*	*WDTREL*	*PCON*	87

Table 7: Special Function Registers Locations

Only a few addresses are occupied, the others are not implemented. SFRs specific to the 652X are shown in **bold** print. Any read access to unimplemented addresses will return undefined data, while any write access will have no effect. The registers at 0x80, 0x88, 0x90, etc., are bit-addressable, all others are byte-addressable.

Special Function Registers (Generic 80515 SFRs)

Table 8 shows the location of the SFRs and the value they assume at reset or power-up.

Name	Location	Reset value	Description
P0	0x80	0xFF	Port 0
SP	0x81	0x07	Stack Pointer
DPL	0x82	0x00	Data Pointer Low 0
DPH	0x83	0x00	Data Pointer High 0
DPL1	0x84	0x00	Data Pointer Low 1
DPH1	0x85	0x00	Data Pointer High 1
WDTREL	0x86	0x00	Watchdog Timer Reload register
PCON	0x87	0x00	UART Speed Control
TCON	0x88	0x00	Timer/Counter Control
TMOD	0x89	0x00	Timer Mode Control
TL0	0x8A	0x00	Timer 0, low byte
TL1	0x8B	0x00	Timer 1, high byte
TH0	0x8C	0x00	Timer 0, low byte
TH1	0x8D	0x00	Timer 1, high byte
CKCON	0x8E	0x01	Clock Control (Stretch=1)
P1	0x90	0xFF	Port 1
DPS	0x92	0x00	Data Pointer select Register
S0CON	0x98	0x00	Serial Port 0, Control Register
S0BUF	0x99	0x00	Serial Port 0, Data Buffer
IEN2	0x9A	0x00	Interrupt Enable Register 2
S1CON	0x9B	0x00	Serial Port 1, Control Register
S1BUF	0x9C	0x00	Serial Port 1, Data Buffer
S1RELL	0x9D	0x00	Serial Port 1, Reload Register, low byte
P2	0xA0	0x00	Port 2
IEN0	0xA8	0x00	Interrupt Enable Register 0
IP0	0xA9	0x00	Interrupt Priority Register 0
S0RELL	0xAA	0xD9	Serial Port 0, Reload Register, low byte
IEN1	0xB8	0x00	Interrupt Enable Register 1
IP1	0xB9	0x00	Interrupt Priority Register 1
S0RELH	0xBA	0x03	Serial Port 0, Reload Register, high byte
S1RELH	0xBB	0x03	Serial Port 1, Reload Register, high byte
USR2	0xBF	0x00	User 2 Port, high address byte for MOVX@Ri
IRCON	0xC0	0x00	Interrupt Request Control Register
T2CON	0xC8	0x00	Polarity for INT2 and INT3
PSW	0xD0	0x00	Program Status Word
WDCON	0xD8	0x00	Baud Rate Control Register (only WDCON.7 bit used)
A	0xE0	0x00	Accumulator
B	0xF0	0x00	B Register

Table 8: Special Function Registers Reset Values

Accumulator (*ACC, A*): *ACC* **is the accumulator register. Most instructions use the accumulator to hold the operand. The mnemonics for accumulator-specific instructions refer to accumulator as "*A*", not *ACC*.**

B **Register:** The *B* register is used during multiply and divide instructions. It can also be used as a scratch-pad register to hold temporary data.

Program Status Word (*PSW*):

MSB   LSB

CV	*AC*	*F0*	*RS1*	*RS*	*OV*	-	*P*

Table 9: *PSW* Register Flags

Bit	Symbol	Function
PSW.7	*CV*	Carry flag
PSW.6	*AC*	Auxiliary Carry flag for BCD operations
PSW.5	*F0*	General purpose Flag 0 available for user. *F0* **is not to be confused with the *F0* flag in the *CESTATUS* register.**
PSW.4	*RS1*	Register bank select control bits. The contents of *RS1* and *RS0* select the working register bank:
PSW.3	*RS0*	<table><tr><th>RS1/RS0</th><th>Bank selected</th><th>Location</th></tr><tr><td>00</td><td>Bank 0</td><td>(0x00 – 0x07)</td></tr><tr><td>01</td><td>Bank 1</td><td>(0x08 – 0x0F)</td></tr><tr><td>10</td><td>Bank 2</td><td>(0x10 – 0x17)</td></tr><tr><td>11</td><td>Bank 3</td><td>(0x18 – 0x1F)</td></tr></table>
PSW.2	*OV*	Overflow flag
PSW.1	-	User defined flag
PSW.0	*P*	Parity flag, affected by hardware to indicate odd / even number of "one" bits in the Accumulator, i.e. even parity.

Table 10: *PSW* Bit Functions

Stack Pointer (*SP*): The stack pointer is a 1-byte register initialized to 0x07 after reset. This register is incremented before PUSH and CALL instructions, causing the stack to begin at location 0x08.

Data Pointer: The data pointer (*DPTR*) is 2 bytes wide. The lower part is *DPL*, and the highest is *DPH*. It can be loaded as two registers (e.g. MOV DPL,#data8). It is generally used to access external code or data space (e.g. MOVC A,@A+DPTR or MOVX A,@DPTR respectively).

Program Counter: The program counter (*PC*) is 2 bytes wide initialized to 0x0000 after reset. This register is incremented when fetching operation code or when operating on data from program memory.

Port Registers: The I/O ports are controlled by Special Function Registers *P0*, *P1*, and *P2*. The contents of the SFR can be observed on corresponding pins on the chip. Writing a '1' to any of the ports (see Table 11) causes the corresponding pin to be at high level (V3P3), and writing a '0' causes the corresponding pin to be held at low level (GND). The data direction registers *DIR0*, *DIR1*, and *DIR2* define individual pins as input or output pins (see section Digital I/O for details).

Register	SFR Address	R/W	Description
P0	0x80	R/W	Register for port 0 read and write operations (pins DIO4…DIO7)
DIR0	0xA2	R/W	Data direction register for port 0. Setting a bit to 1 means that the corresponding pin is an output.
P1	0x90	R/W	Register for port 1 read and write operations (pins DIO8…DIO11, DIO14-DIO15)
DIR1	0x91	R/W	Data direction register for port 1.
P2	0xA0	R/W	Register for port 2 read and write operations (pins DIO16…DIO17, DIO19…DIO21)
DIR2	0xA1	R/W	Data direction register for port 2.

Table 11: Port Registers

All DIO ports on the chip are bi-directional. Each of them consists of a Latch (SFR '*P0*' to '*P2*'), an output driver, and an input buffer, therefore the MPU can output or read data through any of these ports. Even if a DIO pin is configured as an output, the state of the pin can still be read by the MPU, for example when counting pulses issued via DIO pins that are under CE control.

The technique of reading the status of or generating interrupts based on DIO pins configured as outputs, can be used to implement pulse counting.

Special Function Registers Specific to the 71M6521DE/DH/FE

Table 12 shows the location and description of the 71M6521DE/DH/FE-specific SFRs.

Register	Alternative Name	SFR Address	R/W	Description
ERASE	*FLSH_ERASE*	0x94	W	This register is used to initiate either the Flash Mass Erase cycle or the Flash Page Erase cycle. Specific patterns are expected for *FLSH_ERASE* in order to initiate the appropriate Erase cycle (default = 0x00). 0x55 – Initiate Flash Page Erase cycle. Must be preceded by a write to *FLSH_PGADR* @ SFR 0xB7. 0xAA – Initiate Flash Mass Erase cycle. Must be preceded by a write to *FLSH_MEEN* @ SFR 0xB2 and the debug port must be enabled. Any other pattern written to *FLSH_ERASE* will have no effect.
PGADDR	*FLSH_PGADR*	0xB7	R/W	Flash Page Erase Address register containing the flash memory page address (page 0 thru 127) that will be erased during the Page Erase cycle (default = 0x00). Must be re-written for each new Page Erase cycle.
EEDATA		0x9E	R/W	I^{2}C EEPROM interface data register
EECTRL		0x9F	R/W	I^{2}C EEPROM interface control register. If the MPU wishes to write a byte of data to EEPROM, it places the data in *EEDATA* and then writes the 'Transmit' code to *EECTRL*. The write to *EECTRL* initiates the transmit sequence. See the EEPROM Interface section for a description of the command and status bits available for *EECTRL*.

FLSHCRL		0xB2	R/W	Bit 0 (*FLSH_PWE*): Program Write Enable:
				   0 – MOVX commands refer to XRAM Space, normal operation (default).
				   1 – MOVX @DPTR,A moves A to Program Space (Flash) @ DPTR.
				This bit is automatically reset after each byte written to flash. Writes to this bit are inhibited when interrupts are enabled.

			W	Bit 1 (*FLSH_MEEN*): Mass Erase Enable:
				   0 – Mass Erase disabled (default).
				   1 – Mass Erase enabled.
				Must be re-written for each new Mass Erase cycle.

			R/W	Bit 6 (*SECURE*): Enables security provisions that prevent external reading of flash memory and CE program RAM. This bit is reset on chip reset and may only be set. Attempts to write zero are ignored.
			R	Bit 7 (*PREBOOT*): Indicates that the preboot sequence is active.
WDI		0xE8	R/W	**Only byte operations on the whole WDI register should be used when writing**. The byte must have all bits set except the bits that are to be cleared.
			R/W	The multi-purpose register *WDI* contains the following bits: Bit 0 (*IE_XFER*): XFER Interrupt Flag: This flag monitors the XFER_BUSY interrupt. It is set by hardware and must be cleared by the interrupt handler
			W	Bit 1 (*IE_RTC*): RTC Interrupt Flag: This flag monitors the RTC_1SEC interrupt. It is set by hardware and must be cleared by the interrupt handler Bit 7 (*WD_RST*): WD Timer Reset: Read: Reads the *PLL_FALL* interrupt flag Write 0: Clears the *PLL_FALL* interrupt flag Write 1: Resets the watch dog timer
INTBITS	INT0…INT6	0xF8	R	Interrupt inputs. The MPU may read these bits to see the input to external interrupts INT0, INT1, up to INT6. These bits do not have any memory and are primarily intended for debug use

Table 12: Special Function Registers

Instruction Set

All instructions of the generic 8051 microcontroller are supported. A complete list of the instruction set and of the associated op-codes is contained in the 71M6521 Software User's Guide (SUG).

UART

The 71M6521DE/DH/FE includes a UART (UART0) that can be programmed to communicate with a variety of AMR modules. A second UART (UART1) is connected to the optical port, as described in the optical port description.

The UARTs are dedicated 2-wire serial interfaces, which can communicate with an external host processor at up to 38,400 bits/s (with MPU clock = 1.2288MHz). The operation of each pin is as follows:

RX: Serial input data are applied at this pin. Conforming to RS-232 standard, the bytes are input LSB first.

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

The 71M6521DE/DH/FE has several UART-related registers for the control and buffering of serial data. All UART transfers are programmable for parity enable, parity, 2 stop bits/1 stop bit and XON/XOFF options for variable communication baud rates from 300 to 38400 bps. Table 13 shows how the baud rates are calculated. Table 14 shows the selectable UART operation modes.

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

Note: *S0REL* and *S1REL* are 10-bit values derived by combining bits from the respective timer reload registers. *SMOD* is the *SMOD* bit in the SFR *PCON*. *TH1* is the high byte of timer 1.

Table 13: Baud Rate Generation

	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

Table 14: UART Modes

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 *TB80* (*S0CON*.3) and *TB81* (*S1CON*.3) in the *S0COn* and *S1CON* SFRs for transmit and *RB81* (*S1CON*.2) for receive operations. *SM20* (*S0CON*.5) and *SM21* (*S1CON*.5) can be used as handshake signals for inter-processor communication in multi-processor systems.

Serial Interface 0 Control Register (*S0CON*).

The function of the UART0 depends on the setting of the Serial Port Control Register *S0CON*.

MSB   LSB

SM0	*SM1*	*SM20*	*REN0*	*TB80*	*RB80*	*TI0*	*RI0*

Table 15: The *S0CON* Register

Serial Interface 1 Control Register (*S1CON*).

The function of the serial port depends on the setting of the Serial Port Control Register *S1CON*.

MSB   LSB

SM	-	SM21	REN1	TB81	RB81	TI1	RI1

Table 16: The *S1CON* register

Bit	Symbol	Function
S0CON.7	SM0	These two bits set the UART0 mode: | Mode	Description	SM0	SM1
------	-------------	-----	-----
0	N/A	0	0
1	8-bit UART	0	1
2	9-bit UART	1	0
3	9-bit UART	1	1	
S0CON.6	SM1	
S0CON.5	SM20	Enables the inter-processor communication feature.
S0CON.4	REN0	If set, enables serial reception. Cleared by software to disable reception.
S0CON.3	TB80	The 9th transmitted data bit in Modes 2 and 3. Set or cleared by the MPU, depending on the function it performs (parity check, multiprocessor communication etc.)
S0CON.2	RB80	In modes 2 and 3 it is the 9th data bit received. In Mode 1, if *SM20* is 0, *RB80* is the stop bit. In mode 0 this bit is not used. Must be cleared by software
S0CON.1	TI0	Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software.
S0CON.0	RI0	Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software

Table 17: The *S0CON* Bit Functions

Bit	Symbol	Function
S1CON.7	SM	Sets the baud rate for UART1 | SM	Mode	Description	Baud Rate
----	------	-------------	-----------
0	A	9-bit UART	variable
1	B	8-bit UART	variable	
S1CON.5	SM21	Enables the inter-processor communication feature.
S1CON.4	REN1	If set, enables serial reception. Cleared by software to disable reception.
S1CON.3	TB81	The 9th transmitted data bit in Mode A. Set or cleared by the MPU, depending on the function it performs (parity check, multiprocessor communication etc.)
S1CON.2	RB81	In Modes A and B, it is the 9th data bit received. In Mode B, if *SM21* is 0, *RB81* is the stop bit. Must be cleared by software
S1CON.1	TI1	Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software.
S1CON.0	RI1	Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software

Table 18: The *S1CON* Bit Functions

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.