ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Brandywine Communications / JXI2 VME-SyncClock32* at our website: *Click HERE*

VME-SYNCCLOCK32
Synchronized Clock with VME Interface
Operation & Maintenance Manual
7 September 2001 Document No. 8000313
Module S/N_____H/W Rev Level:F1E
Options:_____

jxi2, inc.

10432 Brian Mooney Ave.
El Paso
TX 79935
USA

phone: 915-856-0241
fax: 915-856-0245

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INTRODUCTION 4

SPECIFICATIONS 5
Physical: 5
Inputs & Outputs 5
Bus Interface : 5

INPUT/OUTPUT CONNECTIONS 9

Front Panel J1, J2 (BNC): 9

Front Panel GPS Antenna Input: 9

Front Panel J3 (High density DB26 female): 9

VME Bus Connector (P1) 10

VME Bus Connector (P2) 10

-TAG8 External Time Tag Inputs (P2 or J3) 10

CONFIGURATION, INSTALLATION & OPERATION 11

Identifying Board Options: 11

Configuration: 11
VME Addressing Mode Configuration (P6) 11
Base Address Configuration (P3, P4, P5) 11
Interrupt Level Configuration 11
BNC Connector Configuration (P19) 11
Modulated Code Input Termination (P21) 12
External Oscillator Input Termination (P22) 12
CLK2 output rate source clock selection (P8) 12
GPS Receiver Baud Rate Selection (P13) 12
Standby operation (P18) 12
GPS Antenna/Receiver Power Selection (P20) 12
RS422 output option source selection (P23) 12
1PPS input reference termination (P33) 13

Installation 13
Determining Dual Port Ram Time Code Propagation Delay Setting 13
Determining Dual Port Ram 1PPS Propagation Delay Setting 13

Operation 13
Have Quick 14

LED Status Indication 15

PROGRAMMING 16

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Introduction 16

Reading Current Time (BCD Format) (non -BIN or -TAG8 option) 16

Reading Current Time (BCD Format) (-TAG8 option) 17

Reading Current Time (Binary Format) (-BIN option) 17

Reading Status 17

Reading External BCD Time Tags (-TAG8 option) 18

Reading External BCD Time Tags (non -BIN, non -TAG8 option) 18

Reading External Binary Time Tags (-BIN option only) 19

Simulating External Time Tags (non -TAG8 option) 19

Simulating External Time Tags (-TAG8 option) 19

Reading External Time Tags Using Interrupts (non -TAG8) 19

Reading External Time Tags Using Interrupts (-TAG8 option) 19

Using the RAM FIFO to Buffer External Time Tags 19

Time Match Pulses/Toggles 20

Heartbeat Pulses and Interrupts 20

Dual Port RAM for Supplementary Status/Control 21
- Accessing Dual Port RAM 21
- Sending Commands through the Dual Port RAM. 21
- Setting Leap Year Control in Dual Port RAM 22
- Selecting 1PPS or Time Code Reference Priority in Dual Port RAM 22
- Setting Time Code Propagation Delay Polarity in Dual Port RAM 22
- Setting PPS Propagation Delay in Dual Port RAM 22
- Specifying Modulated Input Code Format in Dual Port RAM 23
- Specifying Leap Second Day in Dual Port RAM 23
- Setting Time Code Propagation Delay in Dual Port Ram 23
- Non-volatile Propagation Delay Settings 23
- Setting PPS Propagation Delay in Dual Port Ram 24
- Setting Major Time in Dual Port RAM 24
- Setting Year in Dual Port RAM 24
- Changing Heartbeat, Lowrate and Rate2 24
- GPS Satellite Tracking and Navigation Data (except Have-Quick receivers) 25
- GPS Satellite Data (Have-Quick receivers) 27

THEORY OF OPERATION 28

Time Code Stripping 28
- Input Signal Coupling 28
- Zero Crossing Detector 28
- Voltage Divider 29

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Peak Detector 29
Burst Decoder 29

FPGA Clock 29
200ms .. 8 Year Bits 29
Disciplining 29
Clock conditioning 30
Modulated code generation 30
Error Measurement 30
Match Times 30
3.0 MHz frequency generation 31
Heartbeat Rates 31

Microcomputer 31
D/A frequency control 31
GPS receiver serial interface 31
HAVE QUICK interface option 31
Power-on reset generation option 31
HAVE QUICK 1 PPS output option 32
-DOSC option (H/W rev E and later -- see mezzanine addendum for H/W rev A-D) 32
LED Time Display 32

VME Bus Interface 32
A16/A24/A32 Address Match 32
Address modes 33
U9, U10 ADRMAT board decode and signal buffers 33
Data buffers 33
IRQL and DTACKL buffers 33
Interrupter 33

BOARD LAYOUT 34

SCHEMATIC DRAWINGS 35

PARTS LIST 39

PROGRAM LISTINGS 41

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Introduction

The VME-SYNCCLOCK32 is a digital clock with a VME bus interface that will automatically synchronize to time reference signals. Programs running on the VME host can read the clock time directly from longword registers (-TAG8 option uses 16 bit registers). In many applications, no other user programming is needed. The advantages of the VME-SYNCCLOCK32's digital clock over the host system "real time" clock include better resolution (100 nsec vs. msec or worse), easier access (single C statement vs. DOS system call) and - most importantly - stability and synchronization to Universal Coordinated Time by locking to time reference signals.

The time reference signals can be standardized modulated "time code" signals (IRIG-B, IRIG-A, NASA-36, or (optionally) IRIG-G,2137 or XR3). Or, the reference can be a 1 PPS time pulse from a GPS or LORAN receiver. Even though the time reference occurs as slowly as once per second, the on-board clock maintains specified accuracy and resolution by using a phase locked 10MHz crystal oscillator. The user program can specify the propagation delay from the time code source to the VME or the 1 PPS source to the VME and the VME-SYNCCLOCK32 will automatically adjust the clock to compensate for the delay. A redundancy feature allows BOTH a 1 PPS signal and a time code to be used as references. The user can specify which reference is preferred. If the preferred reference is invalid, the secondary reference is used until the primary becomes valid again.

In addition to the registers that latch the time when a user program reads the clock, a second set of time registers is used to accurately capture the time of external "time tag" pulses (with 100 nanosecond resolution). Because external pulses may occur at unpredictable times a VME bus interrupt may be enabled by the user to signal the occurrence of an external event and cause the captured time to be stored in host memory, With modern host processor speeds, external pulse spacing as low as a few μseconds may be supported.

The VME-SYNCCLOCK32 is designed so that the most common user applications require the least programming. For example, reading current time can be accomplished with a single C statement. Data and parameters that need fast access and update are assigned individual registers directly accessible over the VME bus, while a much larger number of infrequently (if ever) accessed status and control parameters are accessed by a Dual Port Ram occupying only two VME bus addresses.

Three different types of modular GPS receivers may be plugged into the VME-SYNCCLOCK32 while still occupying a single slot. Or, a user-supplied military GPS receiver (the "Plugger" for example) with "Have Quick" (-HQ option) or 20 bps BCD (-PPTIIN option) output may be used as the time reference. The -TAG8 option supports a total of 8 separate time tag inputs. The -AOUT, -BOUT, and -N36OUT options add modulated IRIG-A,-B or-NASA 36 time code output. The –HQOUT option provides "Have Quick" output. The –DOSC option provides an on-board ovenized oscillator with D/A frequency control with a frequency resolution better than 10^{-10}.
Daughterboard options (VME-SYNCCLOCK32 with daughterboard occupies only 1 VME slot):

- -FIF20,-FIF10 options: 20 or 10 independent external time tags/hardware FIFO buffers.
- -VI option: NTSC/PAL composite video overlay with time (and GPS) data.
- -PARBCD, -PARBIN options: P2 A&C row parallel TTL level BCD or binary output
- -TAG16: 16 separate time tag inputs with independant read register addresses.

Two user selected pulse rates (one from 1.5 Mpps to 45.77 PPS and one configurable 50 PPS to .00153 PPS or 1.5 MPPS to 45.77 PPS) are locked to the clock frequency. The 1.5 M PPS to 45.77 PPS rate can cause "heartbeat" interrupts to periodically wake up host processes. The user also has direct access to a comparator for generating "match" pulses or toggles at precise times.

Source programs and definition files are supplied to speed programming operations.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SPECIFICATIONS

Physical:

Dimensions:	60 mm X 233mm (VME 6U)
Power:	+5V ±5% .50A typ, .75A max
	+12V± 5%: .06A typ .10A max
	-12V± 5%: .025A typ .05A max
Fabrication:	1.68mm±.2mm (.062 in±008in), FR4, 4 layer
Temperature:	0 to 55C operating, -40C to 85C storage
Humidity:	0 to 99% non-condensing

Inputs & Outputs

Inputs:	
Input/output connector:	Hi dens DB26 + Code BNCs
Code input types:	Modulated IRIG-A,B,NASA36. XR3,2137 and IRIG-G optional
Code input amplitude:	250 mV to 10V pk/pk
Code input modulation ratio:	2:1 to 4:1
Code input frequency error:	100PPM maximum
Code input impedance:	>10K ohm
Code Sync accuracy:	1μsec
1PPS Sync accuracy:	100 nsec
External event pulse input:	Rising edge TTL
Minimum event to event spacing (host interrupt mode):	Limited only by host speed
1PPS input:	TTL level positive edge
Have Quick input:	per ICD-GPS-060
RTCM-104 input:	(-RTCM option only) 9600 baud RS422 8 data bits.

Outputs:	
Modulated IRIG-B :	2.5V±10% pk/pk into 600Ω
DC Level Shift IRIG–B:	TTL level
Match:	TTL pulse
Status (Red LED):	Flashes coded patterns

Bus Interface :

Addressing:	A32,A24,A16 A08-A31 user configurable
Data:	D32 time data, D08(O) interrupter/status
Interrupt sources:	4 (Time tag, heartbeat, MATCH, RAM FIFO)
Interrupt levels:	IRQ1 - IRQ7 individually settable
Interrupt vectors:	One for each interrupt source
DTACKL latency:	300 nsec maximum from DSA assert to DTACKL response

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Register Assignments:

Offset from Base Addr	Bits Bit 0 is LSB	Read Usage	Write Usage
1,3,5,7	2.0	same as write	Interrupt channel A..D IRQ level (1-7)
1,3,5,7	3	same as write	Interrupt channel A..D auto clear If set, interrupt request will be cleared automatically when interrupt is acknowledged.
1,3,5,7	4	same as write	Interrupt channel A..D enable
9,B,D,F	7..0	same as write	Interrupt channel A..D vector
11	7..0	RAM FIFO	n/u
13		Status	Read Only
	7	Dual port RAM data ready	Read Only
	6	- FIF20 option: FIFO<19:10> data ready	Read Only
	5	-FIF20 , -FIF10 option: FIFO<9:0> data ready. -TAG16 option: event <15:0> data ready	Read Only
	4	Heartbeat Flag	Read Only
	3	MATCH Flag	Read only
	2	RAM FIFO data available	Read Only
	1	In sync flag	Read Only
	0	Time tag data ready (n/u if TAG8 option installed)	Read Only
15	7..4	Time tag data 800ns..100ns (except -TAG8 option) Time tag 7..0 data ready (-TAG8 option only)	Read Only
17	7..0	n/u	External time tag polarity (chan 7..0)
18	D32	Zero latency (except -TAG8) 40 sec .. 1 μsec BCD time (BCD) μsec<31:0> for -BIN option	Read Only
1C	D32	Zero latency 8 year .. 1 minute BCD time (except -TAG8). For -BIN option: <31:22>=0,<21:14> binary year of century, <13:5> binary day of year, <4:0> = μsec<36:32>	Read Only
18	D16	-TAG8 zero latency 8 msec.. 1 μsec	Read Only
1A	D16	-TAG8 zero latency 80 sec.. 10 msec	Read Only
1C	D16	-TAG8 zero latency 80 hour .. 1 minute	Read Only
1E	D16	-TAG8 zero latency 8 year .. 1 day	Read Only
21	7..0	Dual port RAM data	Dual port RAM data
23	7..0	write only	Dual port RAM address
25	5	n/u	0 = Reset -TAG10, 20 FIFOs
	4	n/u	0 = Reset Heartbeat flag
	3	n/u	0 = Reset Match flag
	2	n/u	0 = Simulate External Event (not -TAG8)
	1	n/u	0 = Release Board Reset
	0	n/u	0 = Set Board Reset

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

27	7..0	n/u	-TAG8 bits<7:0>=Time tag polarity 0= ↑ edge, 1= ↓ edge
29	7..0	n/u	Match time 10μs. 1μs
2B	7..0	n/u	Match time (1 ms, 100 μs)
28	D32	Time tag data latch (except -TAG8) 40 sec .. 1 μsec BCD time μsec<31:0> for -BIN option	n/u
2C	D32	8 years ,, 1 minute BCD time (except -TAG8) For -BIN option: <31:22>=0,<21:14> binary year of century, <13:5> binary day of year, <4:0> = μsec<36:32>	n/u
28	D16	-TAG8 T0 8 msec .. 1μsec	read only
2A	D16	-TAG8 T0 80 sec .. 10 msec	read only
2C	D16	-TAG8 T0 80 hour .. 1 minute	read only
2E	D16	-TAG8 T0 8 year.. 1 day	read only
3x-5x	D32	-FIF20, -FIF10 option registers	n/u
31-3F	D08	-TAG16 option event enable, polarity, & simulated event registers	readonly
30	D16	-TAG8 T1 8 msec .. 1μsec	read only
32	D16	-TAG8 T1 80 sec .. 10 msec	read only
34	D16	-TAG8 T180 hour .. 1 minute	read only
36	D16	-TAG8 T18 year.. 1 day	read only
38	D16	-TAG8 T2 8 msec .. 1μsec	read only
3A	D16	-TAG8 T2 80 sec .. 10 msec	read only
3C	D16	-TAG8 T2 80 hour .. 1 minute	read only
3E	D16	-TAG8 T2 8 year.. 1 day	read only
40	D16	-TAG8 T3 8 msec .. 1μsec	read only
42	D16	-TAG8 T3 80 sec .. 10 msec	read only
44	D16	-TAG8 T3 80 hour .. 1 minute	read only
46	D16	-TAG8 T3 8 year.. 1 day	read only
48	D16	-TAG8 T48 msec .. 1μsec	read only
4A	D16	-TAG8 T4 80 sec .. 10 msec	read only
4C	D16	-TAG8 T4 80 hour .. 1 minute	read only
4E	D16	-TAG8 T4 8 year.. 1 day	read only
50	D16	-TAG8 T5 8 msec .. 1μsec	read only
52	D16	-TAG8 T5 80 sec .. 10 msec	read only
54	D16	-TAG8 T5 80 hour .. 1 minute	read only
56	D16	-TAG8 T5 8 year.. 1 day	read only
58	D16	-TAG8 T6 8 msec .. 1μsec	read only
5A	D16	-TAG8 T6 80 sec .. 10 msec	read only
5C	D16	-TAG8 T6 80 hour .. 1 minute	read only
5E	D16	-TAG8 T6 8 year.. 1 day	read only
60	D16	-TAG8 T7 8 msec .. 1μsec	read only
62	D16	-TAG8 T7 80 sec .. 10 msec	read only
64	D16	-TAG8 T7 80 hour .. 1 minute	read only
66	D16	-TAG8 T7 8 year.. 1 day	read only
71	D08	bits <7:4>: -TAG8 T0 800..100 nsec	read only
73	D08	bits <7:4>: -TAG8 T1 800..100 nsec	read only
75	D08	bits <7:4>: -TAG8 T2 800..100 nsec	read only
77	D08	bits <7:4>: -TAG8 T3 800..100 nsec	read only
79	D08	bits <7:4>: -TAG8 T4 800..100 nsec	read only
7B	D08	bits <7:4>: -TAG8 T5 800..100 nsec	read only
7D	D08	bits <7:4>: -TAG8 T6 800..100 nsec	read only

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7F	D08	bits <7:4>: -TAG8 T7 800..100 nsec	read only
80..FC	D32	-TAG16 Event data registers	read only

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPUT/OUTPUT CONNECTIONS

Front Panel J1, J2 (BNC):

Configurable using P19 as modulated time code input, external oscillator input, or modulated time code output. For units with on-board GPS input, J1 will usually be replaced with a rear bulkhead mount BNC connector for GPS antenna connection. The BNC connector for the GPS antenna has a nut on the front panel and has an non-isolated shield. The J1 connector for non-GPS applications is isolated from the front panel by plastic insulation around the shield.

Front Panel GPS Antenna Input:

For units with on-board GPS receivers, J1 will usually be replaced with a rear bulkhead mount BNC connector for GPS antenna connection. GPS antenna preamp DC power is present on the center pin. If a passive antenna or an antenna shared with other GPS receivers by using a power splitter is used, an external “DC blocker” must be placed between the front panel connector and the antenna or splitter to prevent the preamp DC power from being DC short circuited to ground. The VME-SYNCCLOCK32 has a current limiting device to protect the board, but GPS receiver damage under short conditions is possible. The BNC connector for the GPS antenna has a nut on the front panel and has an non-isolated shield. The J1 connector for non-GPS applications is isolated from the front panel by plastic insulation around the shield.

Front Panel J3 (High density DB26 female):

Pin 1: Time Code Input (“+” side if balanced input)
Pin 2: Time Code Input Ground (or “-” side if balanced input)
Pin 3: Time tag input
Pin 4: Time tag ground return. Connects to ground if optioisolated time tag option is not installed
Pin 5: High rate (1.5 Mpps to 45.7 PPS) user programmable pulse output
Pin 6: “MATCH” pulse output
Pin 7: “Low rate (50 PPS .. .00153 PPS) user programmable pulse output
Pin 8: High rate (1.5M PPS to 45.7 PPS) user programmable HEARTBEAT pulse output
Pin 9: 1 PPS TTL level reference input
Pin 10: Optional modulated time code output
Pin 11: Ground
Pin 12: External oscillator sine wave input
Pin 13: RS422 In + (for Differential GPS RTCM104 input). Should be high in idle state.
Pin 14: RS422 In - (for Differential GPS RTCM104 input). Should be low in idle state.
Pin 15: Ground
Pin 16: Have Quick Serial Data. Connect to pin 7 of Plugger.
Pin 17: Ground. (For Have Quick, connect to pin 2 of Plugger)
Pin 18: Have Quick 1 PPS in. Connect to pin 6 of Plugger.
Pin 19: n/u
Pin 20: Single ended code in
Pin 21: Spare 15 routed to FPGA for future option I/O
Pin 22: Spare 14 routed to FPGA for future option I/O
Pin 23: Spare 13 routed to FPGA for future option I/O
Pin 24: Spare 12 routed to FPGA for future option I/O
Pin 25: Spare 11 routed to FPGA for future option I/O.
Also used for RS422 level IRIG-B output “+” side

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Pin 26: Spare 10 routed to FPGA for future option I/O
Also used for RS422 level IRIG-B output “-” side

VME Bus Connector (P1)

(per VME specification)

VME Bus Connector (P2)

P2 Row A pins 1-32 connect to P12 even numbered pins 2-64.
P2 Row B per VME specification. P2 Row C pins 1-32 connect to P12 odd numbered pins 1-63.

P12 is adjacent to P7 and P7 pins may be connected by wire wrap jumpers to P12 pins for I/O through P2. P7 pins 1-26 carry the same signals as J3 pins 1-26.

-TAG8 External Time Tag Inputs (P2 or J3)

-TAG8 option: P17 pins 45..52 are used for TAG0 .. TAG7 inputs respectively. They may be jumpered to P12 pins for inputs via P2 or to P7 pins for input via J3. Be sure that the connected inputs do not exceed TTL voltage limits (0 to +5V).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CONFIGURATION, INSTALLATION & OPERATION

Identifying Board Options:

A L shaped daughterboard will be installed if the -FIF20, -FIF10, -VI, -TAG16, -PARBCD, or -PARBIN options are installed. The specific daughterboard option can be identified by reading the daughterboard ID next to the daughterboard serial number near the outer daughterboard edge. The -FIF20 and -FIF10 use the same daughterboard (designated FIF20) but the -FIF10 has only half the FIFO devices populated. The -VI option is labeled VME-DVI. The -PARBCD and -PARBIN options both use the -DPAR daughterboard. Users can also telephone or fax the factory (number on cover) to determine what options were installed on the VME-SYNCCLOCK32 when it was shipped, based on the VME-SYNCCLOCK32 serial number.

The -BIN option uses a modified data path FPGA (U18) which will have a programming pattern label (“6812A” for example) attached as well as a label saying “F20BIN”.

The -TAG8 option uses a modified data path FPGA (U18) which will have a programming pattern label (“6B08A” for example) attached as well as a label saying “VME8T”.

Configuration:

The user should check that the configuration of VME-SYNCCLOCK32 matches the configuration that is required by the user application. Factory default configurations are shown below.

VME Addressing Mode Configuration (P6)

The VME-SYNCCLOCK32 may be configured by P6 push on jumpers to operate in A32 mode, A24 mode, A16 mode or all 3 modes:

A16: Jumper P6 “1” to “L” and P6 “0” to “H”
A24: Jumper P6 “1” to “H” and P6 “0” to “L”
A32: Jumper P6 “1” to “H” and P6 “0” to “H”
All: Jumper P6 “1” to “L” and P6 “0” to “L”

Base Address Configuration (P3, P4, P5)

The VME-SYNCCLOCK32 uses 100 (hex) contiguous addresses. Headers with push on jumpers are used to match an address bit as “0” when connected or as “1” when disconnected. Each address bit is marked next to the P3, P4 or P5 header pin used for that address bit.

For A16 address decoding, only A8-A15 need to be configured using P3.
For A24 address decoding, in addition to P3 A16-A23 need to be configured using P4.
For A32 address decoding, in addition to P3 and P4, A24-A31 need to be configured using P5.

Interrupt Level Configuration

Interrupt level for each of the 4 interrupts is set by user programming. See programming section.

BNC Connector Configuration (P19)

P19 configures the J1 and J2 front panel BNC connectors.

The factory configuration without an on-board GPS receiver connects P19-1 / P19-2, P19-3/P19-4, P19-7/P19-8 and P19-9/P19-10 so that J1 is modulated code input and J2 is modulated code output.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The factory configuration with an on-board GPS receiver, the board mounted J1 BNC is removed and replaced with a bulkhead mounted BNC for the GPS antenna input. J2 is still configured for modulated code output. Modulated code input is still possible via J3 or P2.

For external oscillator input on J2, the P19-7/P19-8 jumper is removed and the P19-5/P19-6 jumper is installed. Modulated code output is still possible via J3 or P2.

Modulated Code Input Termination (P21)

Installation of a push on jumper on P21 will terminate the modulated code input with a 51 ohm resistor.

External Oscillator Input Termination (P22)

If J2 is configured for external oscillator input, installation of a push on jumper on P22 will terminate the external oscillator input with a 51 ohm resistor.

CLK2 output rate source clock selection (P8)

P8 configures the source of the user programmable CLK2 divider. Connecting jumper P8-1 ("3M") to P8-2 selects 3 MHz as the input. P8-2/P8-3 ("100") selects 100 PPS as the input.

GPS Receiver Baud Rate Selection (P13)

If the VME-SYNCCLOCK32 has an on-board GPS receiver or communicates with an external GPS receiver serial port, P13 selects the communication baud rate. For 9600 baud, connect P13 pin 2 to P13 pin 1. For 4800 baud, connect P13 pin 2 to P13 pin 3. The factory configuration will be set for the on-board receiver installed. For HAVE-QUICK (-HQ) option, DO NOT INSTALL a jumper on P13.

Standby operation (P18)

The VME-SYNCCLOCK32 has the capability of maintaining clock operation when main VME system power is not present. The microprocessor and FPGA sections of the board are powered by the +5B power bus connected to P18 pin 2. The standard factory configuration connects P18 pin 2 to VME bus +5 power on P18 pin 1. For applications where power down clock operation is required, the VME bus STDBY signal on P18 pin 3 is connected to P18 pin 2.

GPS Antenna/Receiver Power Selection (P20)

If a Trimble or Rockwell GPS receiver module is installed, P20 pin 3 selects the antenna power source. Connect P20 pin 3 to P20 pin 1 for +5V antenna power or to pin 1 for +12V antenna power. Factory configuration is P20 pin 3 to P20 pin 1.

If a GPS receiver is installed, P20 pin 4 selects the +5V receiver power source. To use VME +5 power, P20 pin 4 connects to P20 pin 2. To use VME +5V STDBY power (so that the receiver can remain in operation when the VME system power is off) P20 pin 4 connects to P20 pin 6. Factory configuration is P20 pin 4 to P20 pin 2.

RS422 output option source selection (P23)

If the RS422 output option is installed (U29 pins 5,6) P23 selects the RS422 signal source. For time code DC level shift output, connect P23-2 to P23-1. Connect P23-2 to P23-3 for SPARE12 signal source.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1PPS input reference termination (P33)

The external 1 PPS input (J3 pin 18) and be terminated in 50 ohms by installing a push-on jumper of P33. Do not install the jumper if an on-board GPS receiver is installed. Usually the jumper is only installed for Havequick 1 PPS inputs from a military GPS receiver. For other 1 PPS reference sources, the user should be sure that they can provide a valid TTL level ("1" > 2.4V , 0 < .4V) into a 50 ohm load. If not, the P33 jumper should be left off.

Installation

Double check and record configuration settings before installing the VME-SYNCCLOCK32. The VME system should be powered down before the VME-SYNCCLOCK32 is added. Plug the VME-SYNCCLOCK32 into the appropriate location in the backplane. If the application requires interrupts, be sure the IACKINL signal daisy chain connects to the VME-SYNCCLOCK32 slot. Any empty slots between the IACK source and the VME-SYNCCLOCK32 slot will need to have jumpers connecting IACKOUTL to IACKINL. If a IACKINL to IACKOUTL jumper was installed in the VME-SYNCCLOCK32 slot, the jumper must be removed when the VME-SYNCCLOCK32 is installed.

Attach inputs and outputs before applying power .
Apply power.
Observe LED1 flashing pattern and verify appropriate status per operation notes below.

Determining Dual Port Ram Time Code Propagation Delay Setting

For achieving μsecond accuracy with time code inputs, it is necessary for the user to determine the propagation delay correction for each VME-SYNCCLOCK32 installation. The correct values for propagation delay correction can be estimated base on cable lengths and propagation speed of the cable used, but the effects of capacitive loading usually require that the 10s and units of μsecond digits and below be determined experimentally. Using an approximate value, observe a J3 output that should be "on-time" (for example the negative going edge of IRIGDCOUTL is 100 PPS on-time). Using an oscilloscope, compare the on-time edge with a reference, for example the 1 PPS output of a portable standard or GPS receiver. Another method (if the source is fairly close) is to run **identical length** cables from a pulse output of the time code source and from the pulse output of the VME -SYNCCLOCK32 to the oscilloscope. Adjust the propagation delay settings until the pulses are coincident. Once set, the values are valid unless any changes are made in the connections between the time code source and the VME-SYNCCLOCK32.

Determining Dual Port Ram 1PPS Propagation Delay Setting

For 100 nsec accuracy with the on-board GPS antenna or with an external 1PPS input, it is necessary for the user to set the propagation delay correction for each VME-SYNCCLOCK32 installation. For an external 1 PPS input use a procedure similar to determining the Code Propagation Delay setting. Adjust the value for any source 1 PPS error. For on-board GPS receiver, you can simply set a value based on the antenna cable length.

Operation

The VME-SYNCCLOCK32 will start operation automatically when the VME bus SYSRESETL signal goes low to high. No manual operator intervention is possible or required.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Have Quick

If using a user supplied GPS receiver with HaveQuick output, be sure the receiver setup is configured for Continuous operation with serial output set to STANDARD, 1 PPS set to UTC, and HAVEQUICK set to ON. If your application is a fixed location, setting the GPS receiver to fixed position mode will improve reliability of receiver operation because only a single GPS satellite needs to be in view.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

LED Status Indication

LED1 (red LED) flashes echo the data in the DP_Extd_Sts dual port RAM location. The LSB (DP_Extd_Sts_Nosync) is sent first. “0” bits are short flashes, “1” bits are long flashes. Trailing short flashes are deleted. For example, if DP_Extd_Sts is 1C, the pattern is short short long long long.

red LED	Short	Long
First flash	SynchronizationVerified	Synchronization NOT Verified
Second flash	Time Code Input Readable	Time Code Input NOT Readable
Third flash	1 PPS Verified	1 PPS NOT Verified
Fourth flash	Major Time has been SET after last JAM synchronization	Major Time has NOT been set after last jam synchronization
Fifth flash	Year valid	Year invalid

LED2 (green LED) lights when synchronization has been verified during the last 5 seconds. If a good time code or GPS signal is applied the green LED should light a few seconds later and remain lit after a few more seconds.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

PROGRAMMING

Introduction

The VME-SYNCCLOCK32 is designed so that the most common user applications require the least programming. For example, reading current time can be accomplished with a single C statement. Data and parameters that need fast access and update are assigned individual registers directly accessible over the VME bus, while a much larger number of infrequently (if ever) accessed status and control parameters are accessed by a Dual Port Ram occupying only two VME bus addresses. To allow physical bus address assignments to be assigned to C language variable names, the register variable names are declared as `extern` and the values are linked at link time. The mechanism for defining the addresses for the linker depends on the host software development environment. To compile and link the supplied examples using the Introl C development environment, the VMESYN32.DEF assembler language source file is used to define the assigned address values for the linker. The VMETAG8.DEF source files is used instead of VMESYN32.DEF if the -TAG8 option is installed. The VMEDPDEF.H file contains the Dual Port RAM assignments. The VME32DEF.H file contains the bit assignments for registers.

Sample programs in the appendices show

1. Reading Current Time: if not -TAG8: `curtime.c` if -TAG8: t8`curtim.c`
2. Reading External Time Tags using Interrupts: not -TAG8: `(extintr.c)`, TAG8:`T8INTR.C)`
3. Using the Match register to generate Match pulses at precise times: not -TAG8: `mattime.c`, -TAG8: `t8mattst.c`
4. Using the Dual Port RAM to initialize Major Time for timing using only 1 PPS time reference. `(setmajor.c)`
5. Using the RAM FIFO Port to buffer External Time Tags `(ramfifo.c)`
6. Using the Dual Port RAM to specify a Heartbeat Rate(`heart.c)`

Reading Current Time (BCD Format) (non -BIN or -TAG8 option)

The first read operation (to address Sec40_Usec1 latches all the other current time bits at same time. The clock continues to run and the LATCHED data for all the other digits may be read across the VME bus by following reads. So, Sec40_Usec1 must **always** be read before the other latched time ports are read - even if the user is uninterested in the low order digits. The remaining latched time data may be read in any order. Or, some may be ignored to reduce the amount of stored data or to speed up time reading. The 100 nanosecond through 100 Millisecond bits remain latched until Sec40_Usec1 is read again. The 200 millisecond through 8 years bits remain latched for at least 90 milliseconds after Sec40_Usec1 is read.

Example: Reads current time with two C statements:

```
timelo= Sec40_Usec1;/* sets timelo = SSmmmuuu */
timehi= Year8_Min1; /* sets timehi = YDDDHHMM */
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Reading Current Time (BCD Format) (-TAG8 option)

The first read operation (to address Msec8_Usec1) latches all the other current time bits at same time. The clock continues to run and the LATCHED data for all the other digits may be read across the VME bus by following reads. So, Msec8_Usec1 must **always** be read before the other latched time ports are read - even if the user is uninterested in the low order digits. The remaining latched time data may be read in any order. Or, some may be ignored to reduce the amount of stored data or to speed up time reading. The 100 nanosecond through 100 Millisecond bits remain latched until Msec8_Usec1 is read again. The 200 millisecond through 8 years bits remain latched for at least 90 milliseconds after Msec8_Usec1 is read.

Example: Reads current time with four C statements:

```
timelo= Msec8_Usec1;/* sets timelo = muuu */
timemidlo= Sec80_Msec10;/* Sets timemidlo = SSmm */
timehimid= Hour80_Min1; /* Sets timehimid = HHMM */
timehi= Year8_Day1;     /* Sets timehi = YDDD */
```

Reading Current Time (Binary Format) (-BIN option)

The first read operation (to address Sec40_Usec1) latches all the other current time bits at same time. The clock continues to run and the LATCHED data for all the other digits may be read across the VME bus by following reads. So, Sec40_Usec1 must **always** be read before the other latched time ports are read - even if the user is uninterested in μsec <31:0> bits. The remaining latched time data may be read in any order. Or, some may be ignored to reduce the amount of stored data or to speed up time reading. The 100 nanosecond through 2**18 μsec bits remain latched until Sec40_Usec1 is read again. The binary years, days and μsec <36:19> bits remain latched for at least 90 milliseconds after Sec40_Usec1 is read.

Example: Reads current time with two C statements:

```
timelo= Sec40_Usec1;/* timelo = usec<31:0> */
timehi= Year8_Min1; /* timehi =0 0 0 0 0 0 0 0 0 0 yr<7:0>
                        day<8:0> usec<36:32> */
```

Reading Status

Status bit 0: Ext_Ready bit (READ ONLY) will be “1” if an External Time Tag pulse has occurred since the last read of any of the ExternalTime Tag data registers. It will be cleared when any of the External Time Tag data registers are read across the VME bus.

Status bit 1: The Sync_OK status bit (READ ONLY) will be “1” if the VME clock is synchronized to a time reference input. More extensive time synchronization status information can be read from the dual port RAM DP_Extd_Sts location.

Status bit 2: The RAM_FIFO_Ready status bit (READ ONLY) will be “1” if there is VME-SYNCCLOCK32 FIFO buffered data available. Reading the RAMFifo will clear RAM_FIFO_Ready until the next data byte is available.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Status bit 3: The Match bit (READ ONLY) toggles when the time programmed into Matchlo and Matchhi matches the clock time. The Match latch can be cleared by writing Reset_Match to Resets

Status bit 4: The Heartbeat bit (READ ONLY) is set when a heartbeat pulse occurs at the HBPULSEL output of J3. The heartbeat is generated by dividing a 3 MHz signal by a user programmed divisor of 2 to 65535 (default divisor is 3000 for 1K PPS heartbeat). The Heartbeat latch is cleared by writing Reset_Heartbeat to Resets

Status bit 5 The FIFO70RDY bit is set if the -FIF20 or -FIF10 option is present and there are ANY events in F20_9_0.

Status bit 6: The FIFO1910_RDY bit is set if the -FIF20 option is present and there are ANY events in the F20_19_10.

Status bit 7: The Response_Ready bit is set when Dual Port RAM data is available.

Reading External BCD Time Tags (-TAG8 option)

The -TAG 8 option has a set of time registers for each of the 8 (T7 .. T0) external time tag inputs. Each set of 4 registers may be read directly across the VME bus:

An example reads External Time Tag0 data with 5 C statements:

```
exttimehi= T0_Year8_Day1;      /* sets exttimehi = YDDD */
exttimehimid= T0_Hour80_Min1; /* sets exttimehimid = HHMM */
exttimemidlo= T0_Sec80_Msec10;/* sets exttimemidlo = SSmm */
exttimelo= T0_Msec8_Usec1;     /* sets exttimelo = muuu */
exttimens100= T0_100ns;        /* sets exttimens100 = nnnn */
```

The external time tag latches can be read in any order. and the user can select any or all to read. When any of the external time tag latch ports are read, the corresponding bit in Ext_Rdy is cleared. The 100 Nsec through 100 Millisecond bits remain latched until another external time tag occurs. The 200 millisecond through 8 years bits remain latched for at least 90 milliseconds after the external time tag occurs.

Reading External BCD Time Tags (non -BIN, non -TAG8 option)

The simplest and fastest way to read external time tags uses the VME-SYNCCLOCK32 external time tag latches which may be read directly across the VME bus:

An example reads External Time Tag data with 3 C statements:

```
exttimehi= Ext_Year8_Min1;     /* sets exttimehi = YDDDHHMM */
exttimelo= Ext_Sec40_Usec1;    /* sets exttimelo = SSmmmuuu */
exttimens100= Ext_100ns;       /* sets exttimens100 = nnnn */
```

The external time tag latches can be read in any order. and the user can select any or all to read. When any of the external time tag latch ports are read, the Ext_Ready flag in Status is cleared. The 100 Nsec through 100 Millisecond bits remain latched until another external time tag occurs. The 200 millisecond through 8 years bits remain latched for at least 90 milliseconds after the external time tag occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Reading External Binary Time Tags (-BIN option only)

The simplest and fastest way to read external time tags uses the VME-SYNCCLOCK32 external time tag latches which may be read directly across the VME bus:

An example reads External Time Tag data with 3 C statements:

```
exttimelo= Sec40_Usec1;/* timelo = usec<31:0> */
exttimehi= Year8_Min1; /* timehi =0 0 0 0 0 0 0 0 0 0 yr<7:0>
                              day<8:0> usec<36:32> */
exttimens100= Ext_100ns;            /* sets exttimens100 = nnnn */
```

The external time tag latches can be read in any order. and the user can select any or all to read. When any of the external time tag latch ports are read, the Ext_Ready flag in Status is cleared. The 100 Nsec through 2**18 μsec bits remain latched until another external time tag occurs. The 2**19 μsec through 2**7 years bits remain latched for at least 90 milliseconds after the external time tag occurs.

Simulating External Time Tags (non -TAG8 option)

For program testing purposes the user can simulate the occurrence of an external time tag by writing Trigger_Sim_Ext_Time_Tag to Resets.

Simulating External Time Tags (-TAG8 option)

For program testing purposes the user can simulate the occurrence of any combination of external time tags by writing “0” bits to Ext_Sim. For example to simulate an event for T0 and T1, the data pattern 0xFC would be written to Ext_Sim.

Reading External Time Tags Using Interrupts (non -TAG8)

Reading external time tags at high tag rates is most efficiently accomplished by using the interrupt capabilities of the VME-SYNCCLOCK32. The Time_Tag_Intr_Ctl register is used to set the IRQ level and to enable the time tag interrupt. The Time_Tag_Intr_Vec is used to set an 8 bit interrupt vector which is used by the user’s Interrupt Handler to generate a interrupt vector address when the interrupt occurs. The EXTINTR.C program in the appendix is an example of using External Time Tag Interrupts.

Reading External Time Tags Using Interrupts (-TAG8 option)

Reading external time tags at high tag rates is most efficiently accomplished by using the interrupt capabilities of the VME-SYNCCLOCK32-TAG8. Time_Tag_Intr_Ctl register is used to set the IRQ level and to enable the time tag interrupt. The Time_Tag_Intr_Vec is used to set an 8 bit interrupt vector which is used by the user’s Interrupt Handler to generate a interrupt vector address when the interrupt occurs. In addition, bits in the Ext_IRQ_Enb register must be set to enable or disable interrupt requests from selected external event inputs. For example, to allow interrupts from channel 0 and 1 only, 0x03 would be written to Ext_IRQ_Enb. The T8INTR.C program in the appendix is an example of using External Time Tag Interrupts.

Using the RAM FIFO to Buffer External Time Tags

A second way to read external time tags is to order the VME-SYNCCLOCK32 to store time tag data in an internal RAM First-In First-Out (FIFO) buffer. The RAM FIFO buffer can only be used

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

with the standard external time tag. It does not apply to the additional time tags of the -FIF20 or -FIF10 option. It applies to the T0 channel if the -TAG8 option is installed. The on-board microprocessor will copy a byte at a time from the FIFO to the RAMFifo as the user reads the RAMFifo. The RAM FIFO approach is less efficient because of the byte at a time transfer from microprocessor to RAM FIFO Port and limited in minimum spacing between external events of 250 μsec but may be useful in cases where the user does not wish to use External Time Tag interrupts or polling of the Ext_Ready status bit. The size of the RAM FIFO is 2048 bytes. The default order of data in the RAM FIFO for each external time tag is:

1st Byte:	0xffH synchronizing byte
2nd Byte	External time tag 1μsec, 100 nsec digits
3rd byte	External time tag 100 μsec 10μsec digits
4th byte	External time tag 10 msec, 1 msec digits
5th byte	External time tag 100 msec, 0 digits
6th byte	External time tag 10sec, 1sec digits
7th byte	External time tag 10 min, 1 min digits
8th byte	External time tag 10 hours 1 hour digits
9th byte	External time tag 10 days 1 days digits
10th byte	External time tag 1 years 100 days digits

See the “Sending Commands through the Dual Port RAM” in Dual Port RAM section for enabling RAMFifo time tags. To reduce the transfer of unneeded data and increase the number of time tags that may be stored in the FIFO, the user can change the RAM FIFO First/Count digits in the Dual Port RAM. When the Command_Set_RAM_FIFO command is issued, an unwanted RAM FIFO event may be generated. So, a Command_Clear_RAM_FIFO command should be used to clear the RAM-FIFO right after the Command_Set_FIFO command. Also, the RAM FIFO port should be read to clear any residual data.

Time Match Pulses/Toggles

The Match bit of Status toggles when the time programmed into Matchlo and Matchhi matches the clock time. The Match bit is also output to I/O connectors for use in triggering or gating other equipment. The user may clear the Match bit by writing Reset_Match to Resets. Depending on the application it may be desirable to clear Match right away (for example if the user is trying to generate a set of closely spaced pulses. In other applications (for example generating a logic gate enabling a A/D converter) Match may be allowed to remain set for longer times until another clock value is matched to toggle it off. Before writing new Match register settings, the user should write 0xFF (which cannot be matched because time is in BCD format) into both Match registers to avoid false matches.

Note that the Time Match only applies to 1 μsec through 8 msec bits. It is the user’s responsibility to check for correct upcoming 10 millisecond interval before writing the Match time registers. The Match_Intr_Ctl register controls interrupt enable, auto clear and IRQ level when Match times occur.

Heartbeat Pulses and Interrupts

The Heartbeat bit of the Status will be set when a heartbeat pulse occurs at the HBPULSEL (pin 8) of connector J3. HBPULSEL is driven by the output pulses from a 82C54 programmable counter driven at 3 MHz or 100 PPS as configured by P8. The 3 MHz or 100 PPS is divided by a user controllable divisor of 2 to 65535 (default 3000) to control the heartbeat rate. The Heartbeat_Intr_Ctl register controls interrupt enable, auto clear and IRQ level when HBPULSEL pulses occur.

The heartbeat rate may be changed using the dual port RAM.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Dual Port RAM for Supplementary Status/Control

A 256 byte dual port RAM on the VME-SYNCCLOCK32 shares access between the microprocessor in the VME-SYNCCLOCK32 and the host VME computer. The first 128 locations are READ ONLY from the VME bus. The last 128 locations are READ/WRITE from the VME bus and are initialized after reset by the on-board processor. The dual port address assignments are listed in the VMEDPDEF.H listing in the appendices.

Accessing Dual Port RAM

To reduce the number of bus addresses needed for the dual port RAM, it is accessed over the VME bus in a two step operation. First, the user write the dual port address byte to DPAddr. After checking the Response_Ready flag in Status the current contents of the dual port address selected are available in DPData. To change the current contents of the selected address, the user program writes the new data to the DPData. The write operation is complete when the Response_Ready flag in Status is asserted again. As a double check of correct user programming user may then read DPData and compare the read value with the written value.Remember that the first 128 addresses of the dual port RAM are read-only. Write attempts to the read-only locations will have no effect on the data in the read-only locations

Because accessing the dual port RAM causes interrupts to the microprocessor on the VME-SYNCCLOCK32, the user should limit dual port RAM access rates to less than 2500 per second to avoid interfering with other VME-SYNCCLOCK32 functions. If the VME-SYNCCLOCK32 is being used with a reference input signal with large (over 1000 PPM) frequency error the dual port RAM access rates should be limited to 500 per second. These limitations should not limit the usability of the VME-SYNCCLOCK32 because dual port RAM is used only for data and control/status functions that change infrequently.

Example:

```
/* Specify dual port address */
              DPAddr = DP_Control;
/* wait for VME  to fetch old value */
              while (!(Status &Response_Ready)) {;}
/* Calculate new DP_Control  value */
       new =(DPData ||DP_Control_CodePriority)
/* write new value to DP RAM */
       DPData = new;
/* wait for VME  to acknowledge */
       while (!(Status &Response_Ready)) {;}
       if (DPData != new) {user program error }
```

Sending Commands through the Dual Port RAM.

The DP_Command location in the dual port RAM is used to send commands to the VME-SYNCCLOCK32. Illegal commands will be echoed as 0xFF. The legal commands are:

Command_Set_Major copies time from dual port major time locations to clock
Command_Set_Years copies time from dual port years to clock
(To avoid incorrect clock settings when major time or year has only been partially initialized, the commands signal from the user to the VME that the major time or year is ready to use.)

Command_Set_RAM_FIFO tells the VME-SYNCCLOCK32 to use RAM_FIFO mode for External Time Tags

Command_Reset_RAM_FIFO tells the VME-SYNCCLOCK32 to stop using RAM_FIFO mode for External Time Tags.

Command_Clear_RAM_FIFO tells the VME-SYNCCLOCK32 to set the External RAM

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

FIFO to empty.

Setting Leap Year Control in Dual Port RAM

The DP_Control_Leapyear bit of dual port RAM addressDP_Control_0 tells the VME-SYNCCLOCK32 if the current year is a leap year. Because time code inputs do not contain year information, the VME-SYNCCLOCK32 requires the user to tell the VME-SYNCCLOCK32 if the current year is a leap year. If an on-board GPS receiver is used as the time reference, the leap year information is determined automatically from the year data provided by the GPS receiver. Leap second information is only used by the VME-SYNCCLOCK32 at 365 days 23 hours 59 minutes 59 seconds (or 60 seconds if leap second day =365) to determine if the next day should be 366 (leap year control =1) or 001 (leap year control =0). The DP_Control_Leapyear flag is initialized to 0 when the VME-SYNCCLOCK32 does power -on reset. If set incorrectly, the VME-SYNCCLOCK32 will detect the correct new day about half a time code frame into the new day (.5 seconds for IRIG-B, NASA36, 2137 or .05 seconds for IRIG-A, .005 seconds for IRIG-G) and set the onboard clock time to the correct value. The DP_Control_Leapyear bit is only used if the year in the DP_Year10_Year1 and DP_Year1000_Year100 locations is invalid. If they are valid, the VME-SYNCCLOCK32 can automatically determine leap year status.

If the only reference input to the VME-SYNCCLOCK32 is a 1 PPS pulse it is VERY IMPORTANT to maintain the correct value of the DP_Control_Leapyear bit or to set the year in the DP_Year1000_Year100 and DP_Year10_Year1 locations. If correct values are not maintained, the VME-SYNCCLOCK32 may make an incorrect decision on the day value following Julian day 365.

Selecting 1PPS or Time Code Reference Priority in Dual Port RAM

The VME-SYNCCLOCK32 allows redundant use of a time code input and a 1 PPS (or GPS receiver) input simultaneously as reference. The DP_Control0_CodePriority bit of the DP_Control0 dual port RAM location will be “1” if the time code is the primary reference and “0” if the 1PPS is the primary reference. Default is for 1PPS (GPS) to be the primary reference. The redundancy feature is not available if the GPS receiver is a Rockwell Microtracker LP, because the Microtracker LP time reference pulse output can occur anywhere in a second interval, possibly conflicting with time code time error measurements.

Setting Time Code Propagation Delay Polarity in Dual Port RAM

The VME-SYNCCLOCK32 allows negative propagation delay correction for use in installations where the time code “on-time” position is ahead of on time. Setting the DP_Control0_NegCodePropD bit in the DP_Control0 specifies that the DP_CodePropDel digits specify a NEGATIVE propagation delay correction. DP_Control0_NegCodePropD is cleared on power on or reset.

Setting PPS Propagation Delay in Dual Port RAM

The VME-SYNCCLOCK32 allows negative propagation delay correction for use in installations where the 1 PPS “on-time” position is ahead of on time. Setting the DP_Control0_NegPPSPropD bit in the DP_Control0 specifies that the DP_CodePropDel digits specify a NEGATIVE propagation delay correction. DP_Control0_NegPPSPropD is cleared on power on or reset.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Specifying Modulated Input Code Format in Dual Port RAM

The VME-SYNCCLOCK32 allows the user to specify the format of the modulated input time code in the DP_Code_Select dual port RAM address by writing the following constants defined in VMEDPDEF.H. Default is IRIG-B. If only GPS or 1 PPS is used as a reference, the DP_Code_Select setting is irrelevant.

IRIG-B:	CodeSelect_IRIGB	
NASA36:	CodeSelect_NASA36	
IRIG-A:	Code_Select_IRIGA	
2137:	Code_Select_2137	(requires -R2137 option support)
XR3:	Code_Select_XR3	(requires -RXR3 option support)
IRIG-G	Code_Select_IRIGG	(requires -RG option support)
IRIG-E	Code_Select_IRIGE	(requires -RE option support)

Specifying Leap Second Day in Dual Port RAM

UTC time (Universal Coordinated Time) has an extra second added periodically (about once every 18 months) to keep UTC time aligned with astronomical time. The added second (called a LEAP SECOND) occurs at the end of a specified day so that the last second of that day is at 23 hours 59 minutes 60 seconds. In order for the VME-SYNCCLOCK32 clock to maintain correct UTC time, the VME-SYNCCLOCK32 must know by the PRECEDING second if a given second is a leap second. Warning of upcoming leap seconds is broadcast by GPS satellites starting about 10 days before the leap second occurs. VME-SYNCCLOCK32 units with -GPS option will initialize the Leap Second Day locations if the selected GPS receiver allows the VME-SYNCCLOCK32 to decode it. The standard modulated time codes do not provide leap second advisory information, so the user is allowed to set a value based on user program support. The specified day is usually June 30 or December 31. It is usually announced several months in advance. 10s and 1s of the Julian date are specified in dual port RAM locations DP_Leap_SecDay10Day1 (for 10s and units) and DP_Leap_SecDay1000Day100 for 100s of days. Power on and reset initializes both to FF. For example, say a leap second is to occur on Dec. 31 in a non-leap year. Then the user should set DP_Leap_SecDay10Day1 = 0x65 and DP_Leap_SecDay1000Day100=0x03.

If the Leap Second Day is not set and a leap second occurs, starting with the leap second the VME-SYNCCLOCK32 will be 1 second fast until:

1. If using modulated time code input: about 1/2 way into the next modulated time code field (.5 sec for IRIG-B, NASA36,XR3,2137, .05 sec for IRIG-A, .005 second for IRIG-G)
2. If using PPS reference without time code until a Set Major Time command is issued by the user with the correct time of the most recent PPS pulse set correctly.

Setting Time Code Propagation Delay in Dual Port Ram

DP_CodePropDel-ms100ms10 specifies time code propagation delay 100ms and 10 ms digit.
DP_CodePropDel_ms1us100 specifies time code propagation delay 1ms and 100 μs digits.
DP_CodePropDel_us10us1 specifies time code propagation delay 10μs and 1 μs digits.
DP_CodePropDel_ns100ns10 specified time code propagation delay 100 ns and 10 ns digits.
Power on and reset default setting are 0 for all digits except for units with the -NVP option.

Non-volatile Propagation Delay Settings

Units with the -NVP (non-volatile propagation delay setting) will set the propagation delay to the values automatically stored at the last POWER DOWN. New settings will be stored automatically at POWER DOWN. If using the -NVP option, the user should do a “sanity check” by reading the values after power up from the dual port RAM and checking them for reasonable values.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Setting PPS Propagation Delay in Dual Port Ram

For achieving μsecond accuracy it is necessary for the user to set the propagation delay correction for each VME-SYNCCLOCK32 installation. The correct values for propagation delay correction can be estimated base on cable lengths and propagation speed of the cable used, but the effects of capacitive loading usually require that the 10s and units of μsecond digits and below be determined experimentally. Refer to the installation section of the manual.

DP_PPS_PropDel-ms100ms10 specifies PPS pulse propagation delay 100ms and 10 ms digit.
DP_PPS_PropDel_ms1us100 specifies PPS pulse propagation delay 1ms and 100 μs digits.
DP_PPS_PropDel_us10us1 specifies PPS pulse propagation delay 10μs and 1 μs digits.
DP_PPS_PropDel_ns100ns10 specifies PPS pulse propagation delay 100 ns and 10 ns digits.
Power on and reset default setting are 0 for all digits except for units with the -NVP option. The -NVP option will retain the propagation delay settings at the last POWER OFF.

For achieving best accuracy it is necessary for the user to set the propagation delay correction for each VME-SYNCCLOCK32 installation. Refer to the installation section of the manual.

Setting Major Time in Dual Port RAM

The user MUST initialize Major Time if the 1 PPS input to the VME-SYNCCLOCK32 is the sole timing reference and the seconds, minutes etc. clock times must be correct.. If there is a time code input, the VME-SYNCCLOCK32 can determine major time from the time code. If the -GPS option is used, the VME-SYNCCLOCK32 can determine major time from the GPS receiver serial output. To set major time , the user sets the DP_Major_Time_s10s1 byte for seconds, through DP_Major_Time_d1000d100 for 100s of days digit.

Setting Year in Dual Port RAM

With a time code or PPS reference input, the user can set DP_Year10Year1 and DP_Year1000Year100 for the clock year (10 & 1 digits) to be correct and to allow the VME-SYNCCLOCK32 to automatically detect leap years. If the -GPS option is used, the VME-SYNCCLOCK32 can determine year from the GPS receiver serial output.

Changing Heartbeat, Lowrate and Rate2

The CLK0 signal (J3 pin 7), the Heartbeat signal (J3 pin 8), and the CLK2 signal (J3 pin 5) are generated by three sections (0,1, and 2 respectively) of a programmable 82C54 clock IC.
Each section divides an input signal by a programmable divisor of 2-65535. The input signal for section 0 is a 100 PPS pulse synchronized to the clock. The input signal for sections 2 is a 3 MHz pulse rate also synchronized to the clock. The input for section 1 is configured by P8 to be either 100 PPS or 3 MHz. Because all inputs are synchronized to the clock, they share the same frequency disciplining. So the generated clock outputs will track the reference input like the main clock does. The factory default divisor for CLK0 is 100, so CLK0 will be 1PPS. The factory default divider for section 1(HEARTBEAT) is 3000, so HEARTBEAT will be 1000 PPS. The factory default for CLK2 divider is 39, so RATE2 will be 76.923 KHZ. For -HQ option the CLK2 divider is 555 so CLK2 is about 5.4 KHz. DO NOT change the CLK2 setting for -HQ option or the HaveQuick serial input may be unreadable.

To change the settings from the factory default, the user can change the DP_Ctr2_ctl ,, DP_Ctr0_lmsb values in dual port RAM. A Command_Set_Ctr0, Command_Set_Ctr1 or Command_Set_Ctr2 command can be issued to cause the changed parameters to be changed immediately. or, a Command+Rejam can cause all counters to be set to the new parameters at the start of the next second. The rejam method is used when it is important that a rate not only be synchronized in frequency to the clock, but also with a specific phase. Rejam will force the clock and the counters to resynchronize to the input signal at the start of the next second.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Each of the 3 counters has a control byte (DP_Ctr0_ctl, DP_Ctr1_ctl, DP_Ctr2_ctl) and a 16 bit divide count split into a most significant byte (DP_Ctr0_msb, DP_Ctr1_msb, DP_Ctr2_msb) and a low order byte (DP_Ctr0_lsb, DP_Ctr1_lsb, DP_Ctr2_lsb).

Counter Control Byte Bit Assignments

Bit <0>

Bit value	Mnemonic	Description
1	DP_ctl_bcd	bcd count
0	DP_ctl_bin	binarycount

Bit <3:1>

Bit value	Mnemonic	Description
000	DP_ctl_mode0	“Stop at Terminal Count”. Seldom used. Causes output to go low when Command_Set_Ctrx or Command_Rejam command is issued and to remain low until counter reaches 0. Then output goes high and stays high.
001	DP_ctl_mode1	“Hardware Retriggerable One-Shot”. Seldom used. Output goes low when jam sync occurs and stays low until count=0. Then goes hi until next jam sync.
010	DP_ctl_mode2	“Divide by N Counter”. Out is high until count=0, then low for one CLKx pulse.
011	DP_ctl_mode3	“Square Wave”. Out is high for first half of count and low for second half.
100	DP_ctl_mode4	“Software Trigger”. Seldom used. Output is high until count=0, then low for one CLKx pulse. Sequence is restarted by writing new count.
101	DP_ctl_mode5	“Hardware Triggered Strobe” . Seldom used. Output is high until jam sync. Then after count=0 output goes low one CLKx cycle and then goes high again.

Bit <5:4> should always be 11 (“DP_ctl_rw”)
Bit<7:6>:

Bit value	Mnemonic	Description
00	DP_Ctr0_ctl_sel	Select counter 0
01	DP_Ctr1_ctl_sel	Select counter 1
11	DP_Ctr2_ctl_sel	Select counter 2

GPS Satellite Tracking and Navigation Data (except Have-Quick receivers)

GPS receiver position output data may be accessed through dual port RAM locations defined in `gpsdef.h`. The data is updated once per second. Latitude & longitude data is available in both BCD format (DP_GPS_Lat_MinEM34 .. DP_GPS_Lon_Deg32) and in raw binary format. See `gpsdat.c` for example of using BCD format data. Because the raw binary format does not lose

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

any accuracy in firmware conversion to BCD, it should be used when resolution better than one arc second is required for the application. The raw binary format is GPS receiver type dependant:

For the Motorola Oncore series, the raw binary format is 32 bit twos complement. The value of the LSB is 1 milliarcsecond. Negative values indicate S latitude or W latitude. See `gpsbin.c` for example of using raw binary format data.

Preventing/Correcting Data Change While Reading Position

Once per second the VME-SYNCCLOCK32 firmware takes about 50 milliseconds to receive and convert binary position data arriving from the receiver. The firmware waits until latitude and longitude is received and converted and then copies the converted data in a burst to the dual port RAM. Even though the burst copy only takes a few microseconds every second there is a small possibility that some dual port position byte reads could mix data from two different seconds. Even for static (non moving) applications this could result in wrong data. For example, suppose the latitude is sometimes N 50 degrees 0.00 minutes and sometimes 49 degrees 59.9999 minutes. If data from two different seconds was mixed the user might get 49 degrees 0.00 minutes or 50 degrees 59.9999 minutes. To prevent wrong data the user can:

1. Read a set of lat/lon dual port data and then read a second set. If both sets agree the data is OK. If they don't agree, read a third time. The third set will be OK.

2. As part of an initialization procedure, detect (approximately) when in the second the dual port data is updated. You can determined when the data is updated by doing repeated dual port RAM position reads until a change is detected. The change will always occur at the same point in the second give or take 10 milliseconds. By reading time from the SYNCCLOCK32, the user program can avoid dual port position reads during the update times.

3. Use a semaphore (dual port RAM location DP_GPS_Semaphore) to prevent write/read overlap. The semaphore is 0 after power on reset. Before the firmware copies converted location data to dual port RAM, the firmware decrements the semaphore to attempt a "locked for write" condition. If the result is < 0, the firmware will perform the copy. If the semaphore was >0 before the decrement, a "locked for read" condition prevents the copy. The best way for the user program to lock the DP_GPS_Semaphore for reading is to send the GPS_Readlock command to the dual port DP_Command address. The semaphore value after the (attempted) lock will be returned in the Response port. If it is >0 it is safe to read the dual port RAM. After reading the position data, the user should send the GPS_Readunlock command to the dual port DP_Command address.

4. Test dual port RAM location DP_GPS_UpdateFlag for non 0 (on-board firmware sets non 0 after copying new position data into dual port RAM). New data is available when non 0. After reading data, write 0 into dual port RAM location DP_GPS_UpdateFlag so that it will not be non-0 until next set of new data is written by firmware. See `gpsbin.c` example. The DP_GPS_UpdateFlag is only implemented on units shipped after 25 June 1997.

For compatibility with code written before the lock/unlock commands were added to firmware, the user can read DP_GPS_Semaphore. If <0 copy is under way, so read until 0. If 0, write 1 for "locked for read" condition, read data , and then write a 0 to unlock. This "compatibility" method is not 100% reliable, because there is a small probability that the "lock for write" will be asserted between the "0" read and the "1" write.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

GPS Satellite Data (Have-Quick receivers)

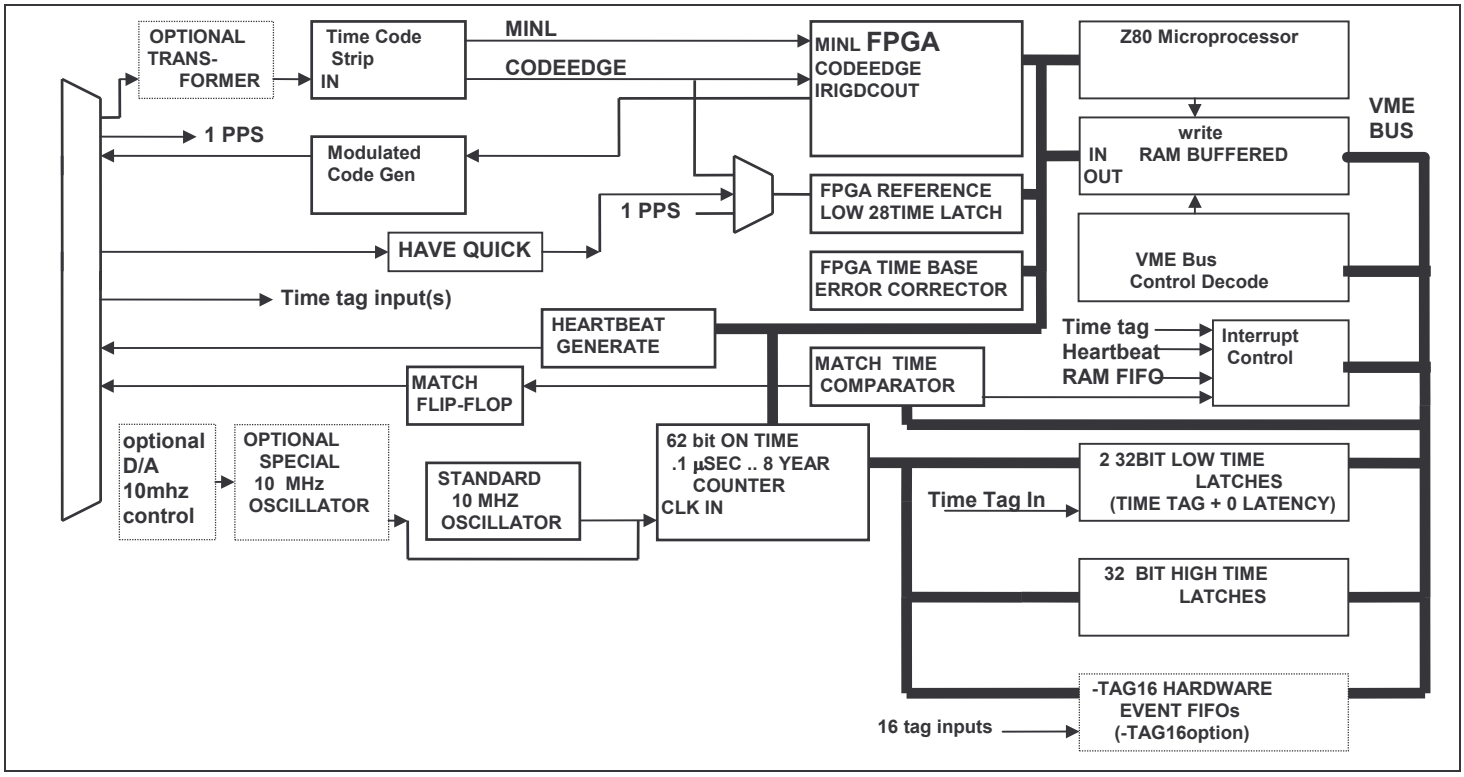
GPS receiver Time Figure-of-Merit may be accessed through dual port RAM locations DP_HQ_TFOM (defined in `gpsdef.h`). The data is updated once per second. The TFOM meaning is:

<0	No information
9	1PPS accuracy no better than 10 milliseconds
8	1 PPS accuracy 1 to 10 milliseconds
7	1 PPS accuracy 100 μsec to 1 millisecond
6	1 PPS accuracy 10 μsec to 100 μsec
5	1 PPS accuracy 1 μsec to 10 μsec
4	1 PPS accuracy .1 μsec to 1 μsec (typical for Plugger)
3	1 PPS accuracy .01 μsec to .1 μsec
2	1 PPS accuracy .001 μsec to .01 μsec
1	1 PPS accuracy better than 1 nsec
0	“Normal” (used when HQ receiver cannot compute actual TFOM)

Because the Have Quick signal does not contain navigation information, only time and TFOM are available.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Theory of Operation



Time Code Stripping

Time code stripping uses a zero crossing detector combined with a voltage divider and a negative peak detector to produce a sampling trigger and a high amplitude cycle pulse to send to a burst decoder in the FPGA device.

Input Signal Coupling

If the “T” option is implemented, transformer T1 couples the isolated CODEIN+ and CODEIN- inputs to the CODEIN and board ground. If no T option, CODEIN+ connects by jumper JPT1-4 to CODEIN and CODEIN- connects by jumper JPT1-3/6 to board ground. Capacitor C1 translates the input signal from + and - voltages around ground to + and - relative to the +5 volts through R2 and RP2-3/4. The input signal impedance is 10.1K ohms (the series resistance of R2 + RP2-3/4).

Zero Crossing Detector

The + side of C1 is the modulated signal input referenced to +5 volts. It is applied to the + side of a LM339 (U22) comparator section at pin U22-5. The - side of the comparator (U22-4) is connected to +5. A 3.3K pull-up resistor pulls up U22-2 whenever U22-5 is HIGHER than U22-4. The open collector output of U22-2 pulls low when U22-5 is LOWER than U22-4. So a +5V/0V square wave in phase with the analog input signal appears at U22-2. Of course there are switching transients showing up as false transitions near the correct zero crossing points that must be dealt with in subsequent processing.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Voltage Divider

The analog reference signal swinging + and - around +5 V is applied to one end of voltage divider R2 and RP2-3/4. The other end of the divider is +5 volts. Because R2 is about twice the value of RP2, the peak to peak AC voltage at the R2/RP2 junction will be 2/3 the peak to peak AC voltage at C1 + side - but still centered on +5 V.

Peak Detector

Because the 2/3 scaled voltage at R2/RP2 junction connects to comparator section U22-9 “+” side input, if U22-9 value is ever LESS than the comparator output voltage U22-14 then U22-14 will be driven lower. This makes U22-14 follow the NEGATIVE peak of the 2/3 scaled signal. A long time constant R3/C2 combination slowly increases the U22-14 voltage if the input signal amplitude decreases. The U22-14 level is the ideal comparison level to separate low and high amplitude input signal cycles. Because “low” amplitude cycles are 1/3 the amplitude of “high” amplitude cycles the 2/3 point is halfway between low and high. U22-14 is applied to U22-10 and compared with the full amplitude signal at U22-11. Whenever the full amplitude signal instantaneous signal is less than the U22-14 level the U22-13 signal will be pulled low, generating negative TTL pulses at about 15% duty cycle.

Burst Decoder

The FPGA includes a precision digital programmable time retriggerable one-shot which triggers on the negative going edge of the zero crossing comparator. The one-shot rearms when the zero crossing comparator output is high. So, the one-shot generates an edge a programmable time delay (about 25 μsec for IRIG-A and 250 μsec for other codes) after the zero crossing negative transition. Noise transitions that have a LOW time less than the programmable time are rejected for noise immunity. By using a counter to generate delay, the temperature varying errors of RC based approaches are avoided. Each delayed edge shifts in the amplitude comparator (U22-13) data for the corresponding cycle of the input code. Logic inside the FPGA detects the beginning of each burst (a burst starts when a large cycle follows a small cycle) and transfers the high/low amplitude information for each cycle of the burst to the microprocessor.

FPGA Clock

The FPGA contains a conventional BCD clock (optional binary) that counts the 10 MHz crystal oscillator pulses. The hardware counter covers the 100 ns bit through the 100 ms bit.

200ms .. 8 Year Bits

Higher order bits (200 ms .. 8 years) are maintained in two sets of latches. One set of latches contains the high order bits for “even” 100 millisecond intervals and the other set contains the high order bits for “odd” intervals. By latching the state of the 100 millisecond counter bit the rest of the high order bits are selected. The high order latches are updated in a leap-frog fashion. Just before the end of a ODD 100 millisecond interval, the EVEN latches are updated to contain correct data for the upcoming even interval. And, just before the end of the EVEN interval, the ODD latches are updated. This approach allows the user to have plenty of time (at least 100 msec) to retrieve time measurements and also allows economical implementation of high order bits.

Disciplining

Disciplining is achieved by periodic advance/retard techniques as used in conventional packaged and board level time code generators. Optionally a voltage controlled crystaloscillator may be

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

installed either at XO1 or on a mezzanine board. When a voltage controlled crystal oscillator is used, a D/A converter is used for fine control of frequency after approximate synchronization is established using the retard/advance circuit. The D/A may be U35 (16 bit 0-10V range), or may be on a mezzaine board if the volatge controlled crystal oscillator is on the mezzanine.

Clock conditioning

Because telecommunications applications usually require that the disciplined oscillator generate a sine wave output, a simple circuit is used to convert the sine wave into a square wave, as well as to protect the VME-SYNCCLOCK32 against transients if an external oscillator is used. C4 AC couples the sine wave and R9/R5 provide hysterisis. CR5 and CR6 provide protection. R10 allows center frequency calibration of oscillators when D/A frequency control is not used.

Modulated code generation

Modulated time code output uses an Exar data book circuit (XR2206 U25) to generate an amplitude modulated sine wave. C7 and RP5-5/6 in series with adjustment pot R8 determine the center frequency (1 KHZ for IRIG-B, NASA36 and 2137, 10 KHZ for IRIG-A, 250 Hz for XR3). The DC frequency control current established by R8/RP5 is augmented by current through RP5-3/4. When the XR2206 is locked to the FPGA clock, the IADJ signal will be a 50% duty cycle 2 KHZ (20 KHZ for IRIG-A, 500 Hz for XR3) square wave. If the XR2206 is running fast, the square wave will have a lower duty cycle, reducing the augmented current and reducing the XR2206 frequency. If the XR2206 is running fast, the square wave duty cycle will increase, raising the XR2206 frequency. RP5-1/2 and C6 filter out the AC component of the square wave leaving a slowly varying DC voltage at C5 for XR2206 frequency control. The sine wave output at U25 pin 2 is amplitude modulated using the AM input pin 1. The 0/5V IRIGDCOUTL signal is LOW for large amplitude bursts and HIGH for small amplitude bursts. The 0/5 is scaled by voltage divider R6/R7 to 2.5/5 V. Trimmer pot R13 allows adjustment of the modulated sine wave symmetry. Because the modulated sine wave output is riding on a +6V DC signal, it is AC coupled to an output driver U26 by capacitor C8 and drain resistor R14. The time code output at the input to driver U26 is 2.5V pk/pk. U26 uses a non-inverting feedback configuration for a gain of 2X. U26 is capable of driving 50 ohm loads, and uses series resistor RP6-3/4 to set a source resistance of 51 ohms. When terminated at the receiving system in 50 ohms, the 2.5V pk/pk value will be received.

Error Measurement

The FPGA contains internal “reference” latches for capturing the time of the clock when the one-shot sampling pulses occur (or one sample per burst for high speed codes like IRIG-A). The on-board microprocessor uses the captured times to estimate the difference between the clock time and the modulated code time. If the time code input has priority for error correction, the difference information from the burst capture is used for disciplining. The VME-SYNCCLOCK32 firmware can also capture the time of a reference 1PPS input. By sharing the reference latches between the between time code and 1 PPS pulses, redundant operation is possible.

Match Times

The FPGA contains two 8 bit comparators directly accessible across the VME bus to allow the user to toggle the MATCH flip flop when the user programmed times match the clock time. Direct bus access and an associated interrupt allow user’s to generate very tightly spaced pulses at arbitrary times that cannot be achieved in conventional approaches which maintain the comparators under microcomputer control.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.0 MHz frequency generation

3.0 MHz synchronous to the main clock is output by the FPGA for use in generating “modem” rates 24000 Hz, 2400 Hz etc. that are not easily derived from 10 MHz or 5 MHz. The 3 MHz is disciplined by the same mechanisms used for the FPGA clock, so it will be synchronous to the FPGA clock.

Heartbeat Rates

A conventional Intel architecture 82C54 triple 16 bit counter (U23) is used to generate three user controlled rates / frequencies. The rates and frequencies will track the on-board clock because they are generated from the same disciplining clock frequencies.

Microcomputer

The microcomputer section is a conventional Z80 microprocessor (U21), 8K X 8 EPROM (U20), 8K X 8 RAM (U19) and triple programmable counter (U23). The dual port RAM accessed by the host program is maintained in part of U19 by U21 control. The microprocessor runs at 5 MHz derived from XO1 (10 MHz) divided by 2 inside FPGA (U11). The FPGA U18 generates Z80 interrupts for code reading, disciplining, external time tag processing (if user enabled), and ODD.EVEN latch updating and dual port RAM support.

D/A frequency control

U35 is controlled by sending data bits to it in serial accompanied by a SERCLK clock signal. The data bits come directly from the Z80 data bus and the SERCLK signal is generated by decoding an I/O address in U17. The SERLDL signal transfers the data to the D/A converter and is also generated by I/O address decoding in U17.

GPS receiver serial interface

A Z80 SIO (U24) allows 4800 baud or 9600 baud serial bi-directional communication between an on-board GPS receiver and the Z80 microcomputer using channel A of the two channel SIO. The 32X baud rate clock is generated by a divider implemented in PAL device U31 and driven by a 1.8432 MHz oscillator XO2. Channel B of the SIO is configured for 4800 baud reception of RTCM104 format serial differential GPS signals for cases where the Motorola VP Oncore GPS receiver is used.

HAVE QUICK interface option

The HAVE QUICK GPS receiver interface receives the 1 PPS (PTTI1PPS) and serial data (PTTI) from a user furnished GPS receiver (example: PLUGGER). The 1PPS is clamped by RMOD3 and CR1 to TTL levels and used as the 1 PPS reference by FPGA U18. The PTTI data is input to the RXDB pin of U24. Raw HaveQuick serial Manchester data is sampled at 5.4KHz by U24 and converted by firmware into binary data.

Power-on reset generation option

Some optional high stability oscillators require several seconds after power-on to start oscillating. For these oscillators, a TL7705A power supervisor (U30) with a power-on delay timing capacitor C27 will be installed. The TL7705A output is wire-ored with the open collector RESETL signal generated by U11. If the high stability oscillator is on a mezzanine module, the power-on delay may be implemented on the mezzanine.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

HAVE QUICK 1 PPS output option

IC U39 converts a 20 μsec TTL pulse to a (Thevenin equivalent) 10V pulse with a 50 ohm source impedance. The source impedance is established by a section of RP6.

-DOSC option (H/W rev E and later -- see mezzanine addendum for H/W rev A-D)

The 16 bit D/A converter U35 (AD660) is used to control the frequency of 10 MHz ovenized crystal oscillator XO1, XO1ALT, or XO1ALT2. The D/A converter scales the REF output of the D/A converter under control of the microprocessor on the main. The REF output of the oscillator is buffered by precision op amp U36 (OP27). Because high stability crystal oscillators have slow start up times (as long as 5 seconds), a reset circuit U30/C27 extends the microprocessor RESETL signal on power up to allow the oscillator to start.

LED Time Display

A nine character display is used for displaying time in DDD HH MM SS format. The LED segments are arranged in an 8 X 8 matrix. The digit selects for 10 days .. 1 second digits form the "columns" of the matrix. The segment selects for the 7 segment display sections a..g form 7 rows of the matrix. A different column is selected by U21 (as decode by U17 to form the DIGIT signal) every millisecond and the U28 output for that column ONLY is written into 27 (as decoded by U17 to form the SEGMENT signal) is asserted low at the output of U28. At the same time, the segment drives LA..LG that should be driven HIGH for the selected character are driven during that millisecond. Resistor networks RP3 and RP4 limit segment drive currents. The LED devices for 10 days .. 1 second are common CATHODE so that a digit is selected when it's digit drive on connector P15 is LOW and the segments for that digit are illuminated when they are driven HIGH at connector P16.

A trick is used to implement the 100 days digit display. Each of it's seven segments looks to the LED drive circuits like the decimal point segment of the seven other DIGITS. It uses a common ANODE display circuit, because the common anode will be driven HIGH by the LDP segment select and each segment will be selected when the corresponding DIGIT drive signal is LOW.

Firmware converts the BCD time of day into the 64 matrix values at the start of each second. Because the column select rate is 1000/second, and their are 8 columns, each digit gets driven 1/8th of the time at a repetition rate of 125 times/second.

Because the LED display is multiplexed at a 12.5 % duty cycle for each digit, each digit is driven with a current in excess of the allowable DC value. To protect the display from burning out a digit in event of a firmware crash or extended SYSRESET, the DIGITL pulses drive a RC timer R18/C13. If DIGITL pulses disappear for more than 1/2 second, C13 will discharge to ground through R18, driving U32 pin 11 low and U28 pin 22 high, disabling digit output.

VME Bus Interface

A16/A24/A32 Address Match

Three 8 bit identity comparators are used to match VME bus addresses with the address configured in P3, P4 and P5. Based on the setting of P6 (see configuration section), PAL device U8 requires:

A16: comparator U1 output A1508EQL be low
A24: comparator U1 output and U2 output (A2316EQL) be low.
A32: comparator U1 output, U2 output and U3 output (A2123EQL) be low.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Address modes

For A16 accesses (if enabled) U9 decodes only AM<5:0> = 29 or 2D
For A24 accesses (if enabled) U9 decodes only AM<5:0> = 39 or 3D
For A32 accesses (if enabled) U9 decodes only AM<5:0> = 09 or 0D

U9, U10 ADRMAT board decode and signal buffers

If the address mode is decoded and the required address comparator outputs are low and IACKL is not asserted, PAL device U9 will output ADRMAT high. U9 also buffers the VME bus DS0L data strobe and the VME bus WRITE signals so that they draw only a single bus load. U10 buffers VME bus address bits A1 through A6, SYSCLK and LWORDL for distribution to FPGA devices U11 and U18.

Data buffers

For byte transfers (LWORDL not asserted) VME bus data bits 0-7 are buffered for reads & writes and interrupt vectors by U6. For D32 (longword) reads, buffers U4, U8 and U9 are enabled. Because the -TAG8 option uses D16 transfers for time transfer, U8 and U9 are always disabled (/enable inputs driven high) so LD16 - LD23 can be used as TAG0-TAG7 inputs.

IRQL and DTACKL buffers

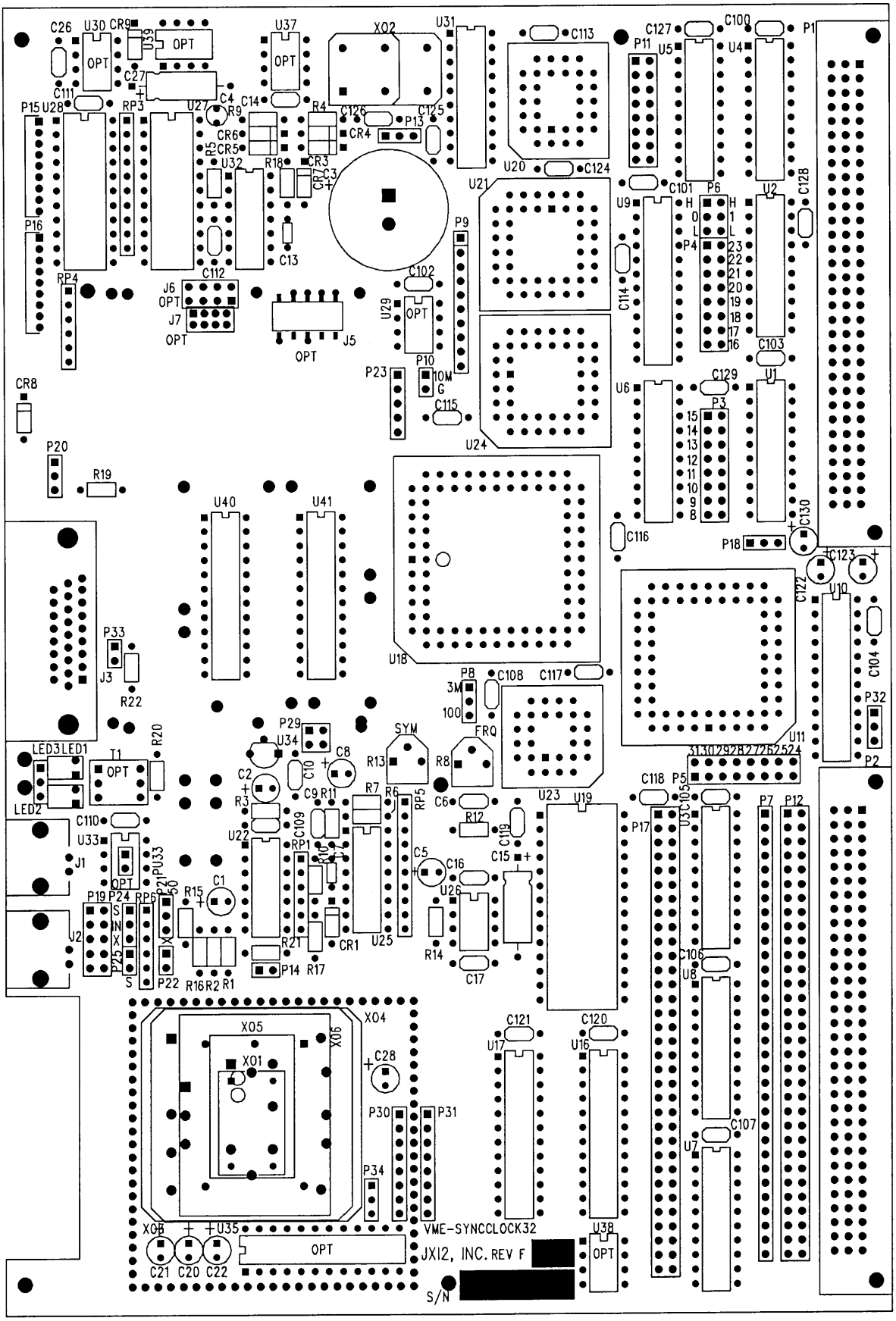
An eight section open collector buffer 74LS641-1 device (U7) drives IRQ1L - IRQ7L if the respective inputs are asserted low by U11 outputs. DTACKL is asserted low if the wire-ORed signal LDTACKL (RP1 pin 5 3.3K pull-up to +5) is pulled low by U11 output or U18 output.

Interrupter

FPGA device U11 implements a 4 channel interrupter emulating the classic MC68153. It also supports address decoding for the daughter FIFO devices for the -TAG16 option. Finally, it contains interrupt request trigger inputs:

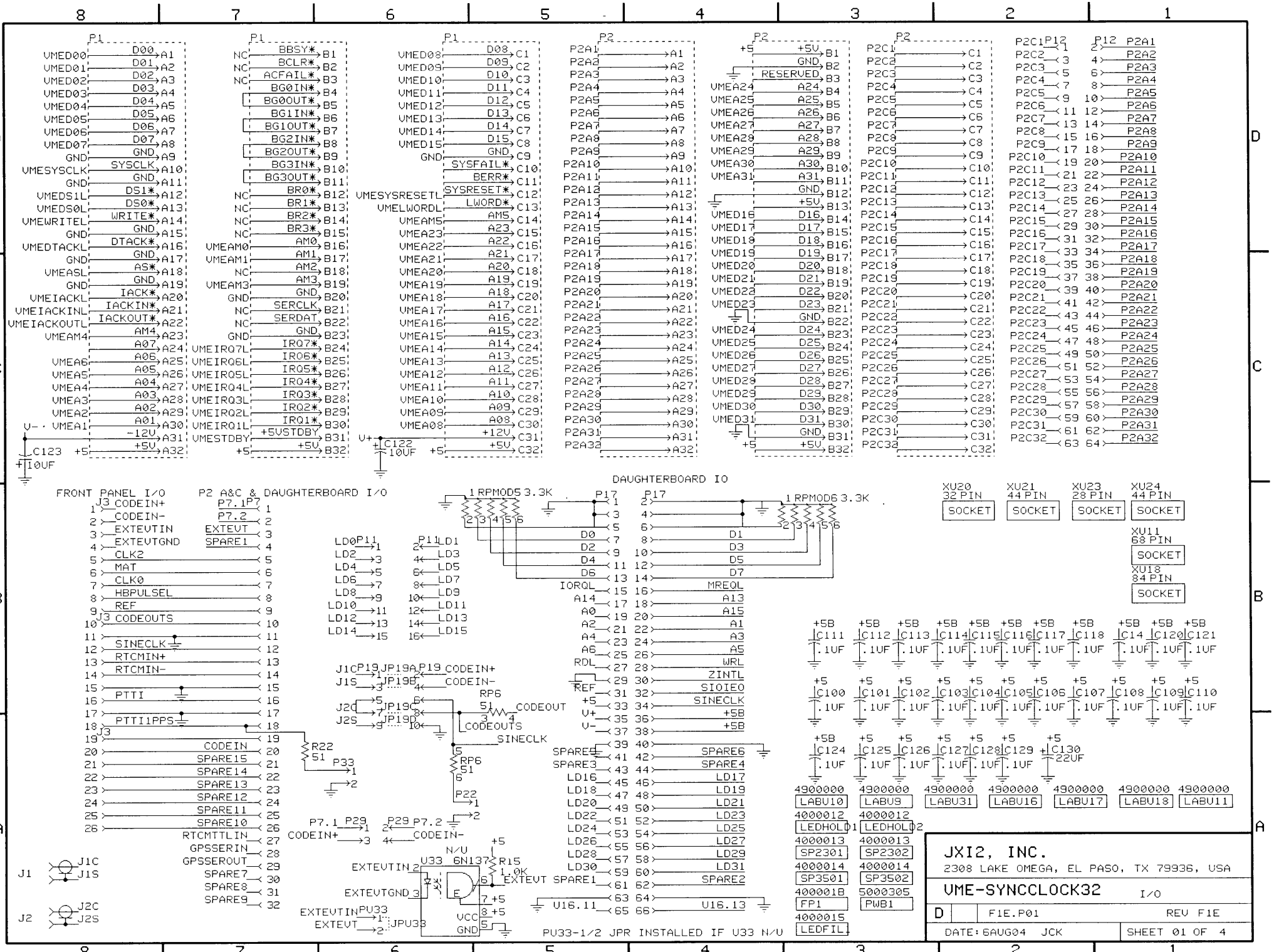
Heartbeat interrupts:	HBPULSEL
Time tag interrupts:	EXTEVT
Time “MATCH” interrupts:	MAT
RAM FIFO data ready interrupts:	RAMFIWR

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

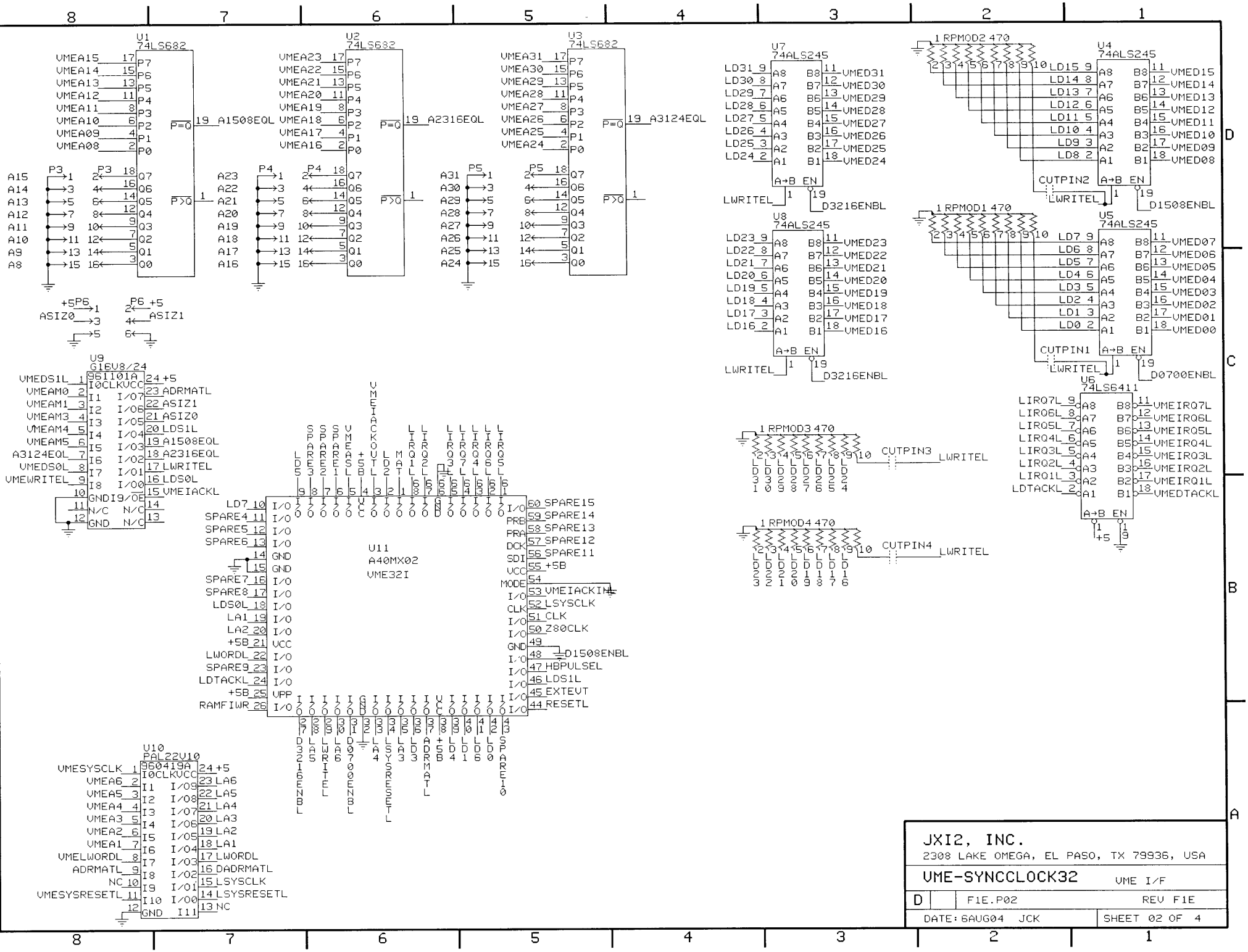


SILKSCREEN-TOP

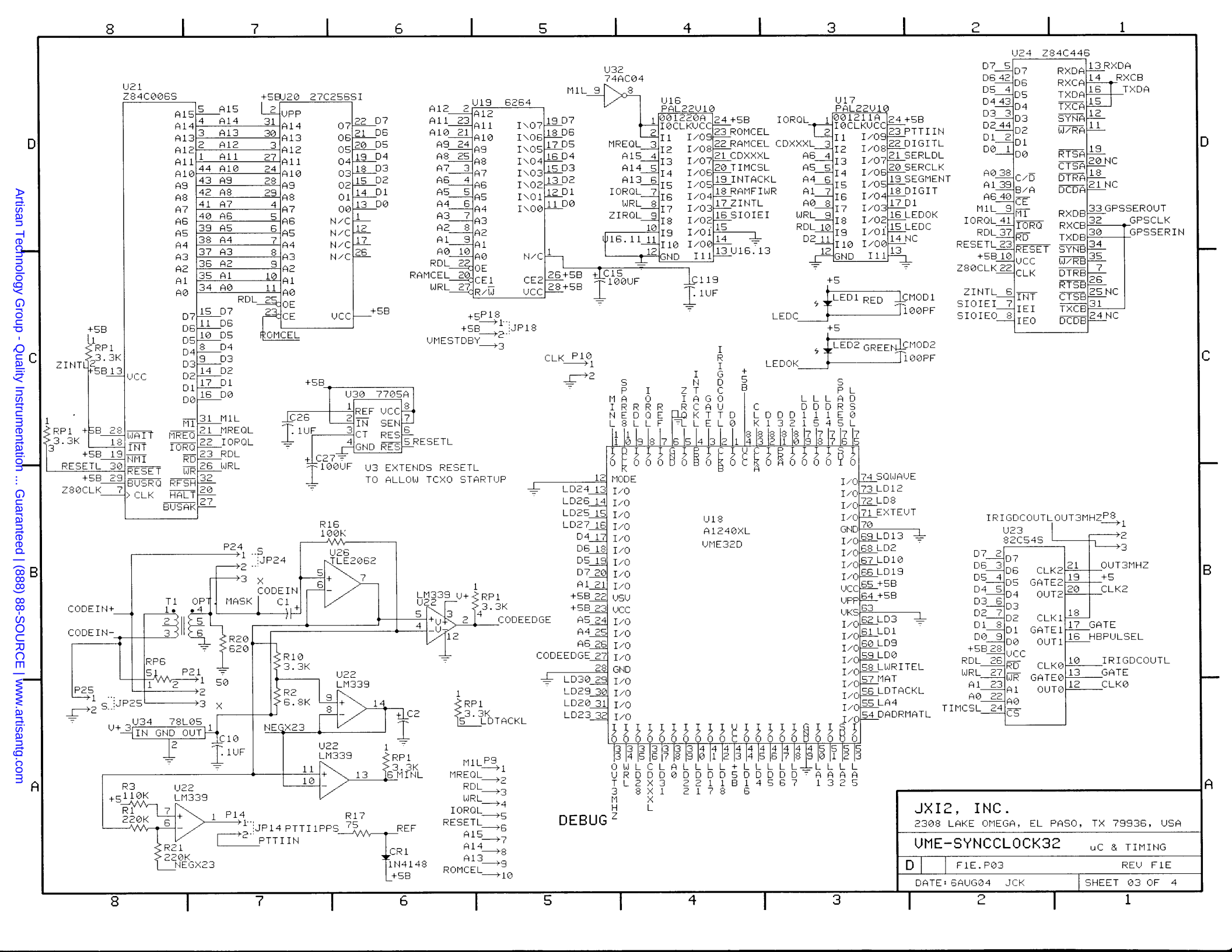
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



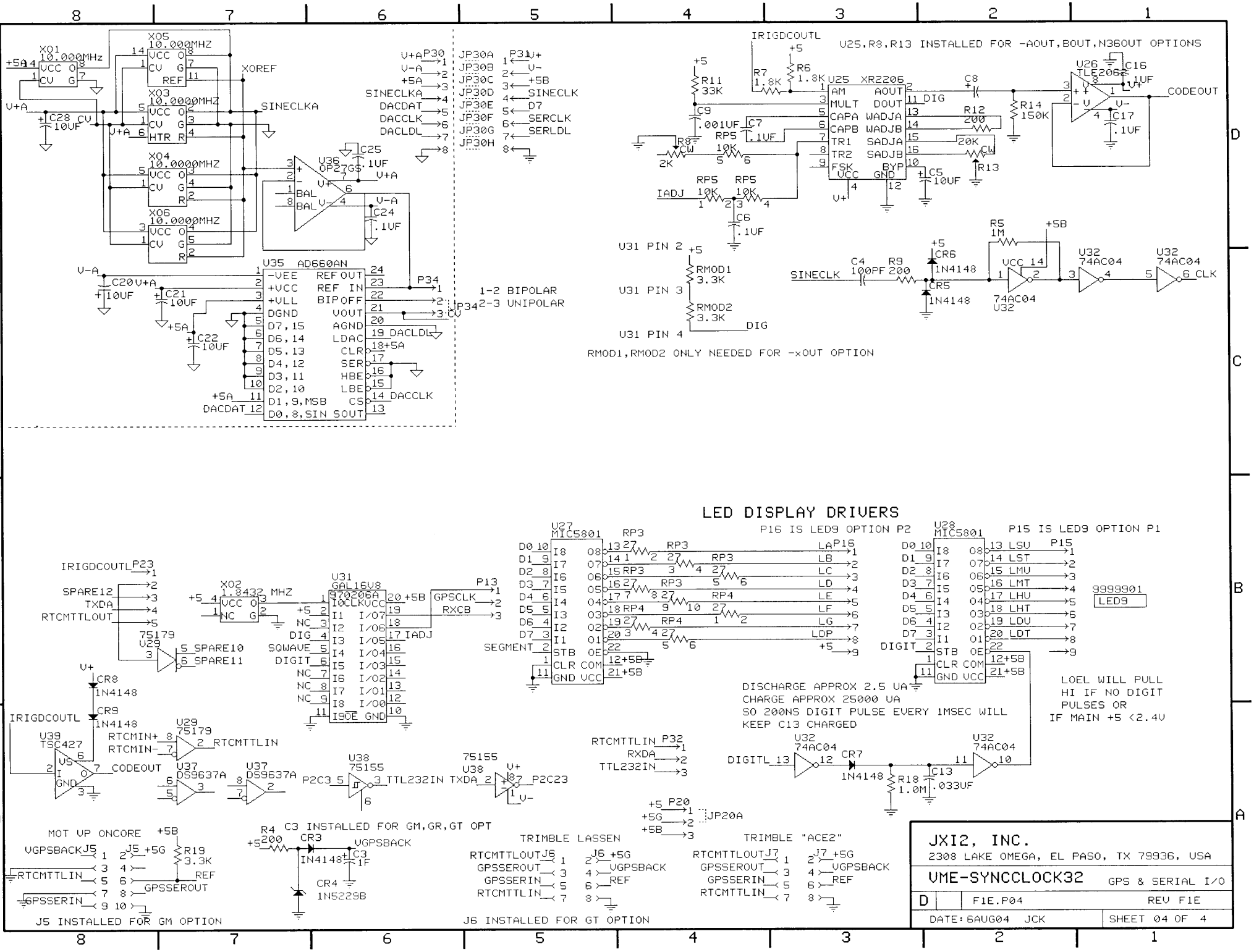
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

QTY	JXN	DESCRIPTION	COMMENTS	REF. DES.
1	0	JUMPER, SOLDERED (SOLID) P14-1/2		JP14
1	0	JUMPER, SOLDERED (SOLID) P18-1/2		JP18
1	0	JUMPER, SOLDERED (SOLID) P19-1/2		JP19A
1	0	JUMPER, SOLDERED (SOLID) P19-3/4		JP19B
1	0	JUMPER, SOLDERED (SOLID) P19-7/8		JP19C
1	0	JUMPER, SOLDERED (SOLID) P19-9/10		JP19D
1	0	JUMPER, SOLDERED (SOLID) P20-1/2		JP20A
1	0	JUMPER, SOLDERED (SOLID) P24-1/2		JP24
1	0	JUMPER, SOLDERED (SOLID) P25-1/2		JP25
1	0	JUMPER, SOLDERED (SOLID) P34-2/3	OPT.	JP34
1	0	JUMPER, SOLDERED (SOLID) PU33-1/2		JPU33
1	0000000	MOD.-PIN CUT: RPMOD1-10	*	CUTPIN1
1	0000000	MOD.-PIN CUT: RPMOD2-10	*	CUTPIN2
1	0000000	MOD.-PIN CUT: RPMOD3-10	*	CUTPIN3
1	0000000	MOD.-PIN CUT: RPMOD4-10	*	CUTPIN4
3	0000001	IC, 74LS682 COMPARATOR 8 BIT INT. PULLUP		U1,U2,U3
1	0000002	IC, 74LS641-1 OCTAL O.C. DRVR 48MA.		U6
1	0000300	IC, 74AC04 HEX INVERTER -40/+85C		U32
1	0020000	IC, 6264 8K X 8 RAM,150NS		U19
1	0030000	IC, LM339 QUAD VOLTAGE COMPARATOR		U22
1	0030002	IC, XR2206 LINEAR WAVEFORM GENERATOR	OPT MASK	U25
2	0030004	IC, UCN5801A OCTAL LATCHED DRIVER		U27,U28
1	0030005	IC, TL7705A 4.75V RESET SUPERVISOR	OPT. MASK	U30
1	0030014	IC, 78L05 5V VOLTAGE REGULATOR		U34
1	0030022	IC, TLE2062 HI DRIVE/LO OFF DUAL OP AMP		U26
1	0030027	IC, TSC427 MOSFET DRIVER	OPT. MASK	U39
1	0040002	IC, 75179 RS422 DRIVER/RECEIVER	OPT MASK	U29
1	0040005	IC, AD660AN 16 BIT DAC V OUT	OPT.	U35
1	0040008	IC, 75155 RS232 DRIVER/RECEIVER	N/U MASK	U38
1	0040010	IC, DS9637ACN DUAL RS422 RECEIVER	OPT. MASK	U37
1	0090000	IC, GAL 16V8, 961101A		U9
1	0090000	IC, GAL 16V8, 970206A		U31
1	0090002	IC, PAL 22V10, 001211A		U17
1	0090002	IC, PAL 22V10, 001220A		U16
1	0090002	IC, PAL 22V10, 960419A		U10
7	0100001	DIODE, 1N4148 SILICON SIGNAL		CR1,CR3,CR5,CR6,CR7, CR8,CR9
1	0100005	DIODE, ZENER 1N5229B 4.3V VZ 20MA IZ		CR4
1	0120003	LED, RED T1 (INT. RES)		LED1
1	0120004	IC,OPTOISOLATOR 6N137 TTL COMPATIBLE	OPT MASK	U33
1	0120005	LED, GREEN T1 (INT. RES)		LED2
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-1/P31-1		JP30A
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-2/P31-2		JP30B
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-3/P31-3		JP30C
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-4/P31-4		JP30D
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-5/P31-5		JP30E
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-6/P31-6		JP30F
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-7/P31-7		JP30G
1	0200000	JUMPER, 0 OHM 1/8W RESISTOR P30-8/P31-8		JP30H
1	0201510	RESISTOR, 51 OHM 1/8W 5% CF		R22

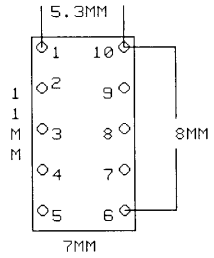
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

QTY	JXN	DESCRIPTION	COMMENTS	REF. DES.
1	0201750	RESISTOR, 75 OHM 1/8W 5% CF		R17
3	0202200	RESISTOR, 200 OHM 1/8W 5% CF		R4,R9,R12
1	0202620	RESISTOR, 620 OHM 1/8W 5% CF		R20
1	0203100	RESISTOR, 1K OHM 1/8W 5% CF		R15
2	0203180	RESISTOR, 1.8K OHM 1/8W 5% CF		R6,R7
2	0203330	RESISTOR, 3.3K OHM 1/8W 5% CF		R10,R19
1	0203680	RESISTOR, 6.8K OHM 1/8W 5% CF		R2
1	0204330	RESISTOR, 33K OHM 1/8W 5% CF		R11
1	0205100	RESISTOR, 100K OHM 1/8W 5% CF		R16
1	0205110	RESISTOR, 110K OHM 1/8W 5% CF		R3
1	0205150	RESISTOR, 150K OHM 1/8W 5% CF		R14
2	0205220	RESISTOR, 220K OHM 1/8W 5% CF		R1,R21
2	0206100	RESISTOR, 1.0M OHM 1/8W 5% CF		R5,R18
1	0279320	POTENTIOMETER, 2K OHM "P" CASE	OPT. MASK	R8
1	0279420	POTENTIOMETER, 20K OHM "P" CASE	OPT. MASK	R13
1	0283127	RESISTOR NET, 27 OHM 3 SEP RES 6 PIN SIP		RP4
1	0283151	RESISTOR NET, 51 OHM 3 SEP RES 6 PIN SIP		RP6
1	0285127	RESISTOR NET, 27 OHM 5 SEP RES 10 PIN SIP		RP3
1	0285410	RESISTOR NET, 10K OHM SEPARATE 10 PIN SIP		RP5
1	0286333	RESISTOR NETWORK, 3.3K OHM BUSSED 6 PIN SIP		RP1
1	0286333	RESISTOR NETWORK, 3.3K OHM BUSSED 6 PIN SIP	* P17-3	RPMOD5
1	0286333	RESISTOR NETWORK, 3.3K OHM BUSSED 6 PIN SIP	* P17-4	RPMOD6
1	0289247	RESISTOR NETWORK, 470 OHM BUSSED 10 PIN SIP	* U4-10	RPMOD2
1	0289247	RESISTOR NETWORK, 470 OHM BUSSED 10 PIN SIP	* U5-10	RPMOD1
1	0289247	RESISTOR NETWORK, 470 OHM BUSSED 10 PIN SIP	* U7-10	RPMOD3
1	0289247	RESISTOR NETWORK, 470 OHM BUSSED 10 PIN SIP	* U8-10	RPMOD4
1	0321000	CAPACITOR, 100PF 5% NPO MLC		C4
1	0331000	CAPACITOR, .001UF 5% NPO MLC AXIAL		C9
1	0343300	CAPACITOR, .033 UF 5% METAL FILM		C13
1	0351000	CAPACITOR, .1UF 5% METAL FILM		C7
34	0351001	CAPACITOR, .1UF Z5U MLC		C6,C10,C14,C16,C17,C26 C100,C101,C102,C103 C104,C105,C106,C107 C108,C109,C110,C111 C112,C113,C114,C115 C116,C117,C118,C119 C120,C121,C124,C125 C126,C127,C128,C129
7	0371000	CAPACITOR, 10UF 16V TANTALUM		C1,C2,C5,C8,C22,C122 C123
3	0371000	CAPACITOR, 10UF 16V TANTALUM	OPT.	C20,C21,C28
1	0372200	CAPACITOR, 22UF 10% 6V TANTALUM		C130
1	0381001	CAPACITOR, 100UF 20% 6V AXIAL ELECTROLYTIC		C27
1	0381001	CAPACITOR, 100UF 20% 6V AXIAL ELECTROLYTIC	N/U	C15
1	0399001	CAPACITOR, 1F "GOLD" BACKUP POWER	N/U MASK	C3
1	0400000	TRANSFORMER, 10K:10K	OPT. MASK	T1
1	0500002	OSCILLATOR, CRYSTAL 10.000MHZ		XO1
1	0500007	OSCILLATOR 1.8432 MHZ	OPT. -GM	XO2
1	0500015	OSCILLATOR OCXO 10.0000MHZ LO PROFILE	N/U	XO3
1	0500016	OSCILLATOR OCXO 10.0000MHZ 240 SERIES	N/U	XO4
1	0500017	OSCILLATOR OCXO 10MHZ AIOC1500SMH	N/U	XO5
1	0500024	OSCILLATOR OCXO 10.0000MHZ LO PROFILE	N/U	XO6
2	0700002	CONNECTOR, 1 X 2 .025 SQ HEADER		P10,P33
3	0700002	CONNECTOR, 1 X 2 .025 SQ HEADER	N/U	P14,P25,PU33

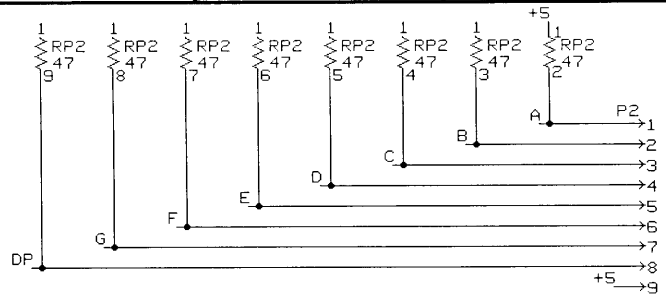
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

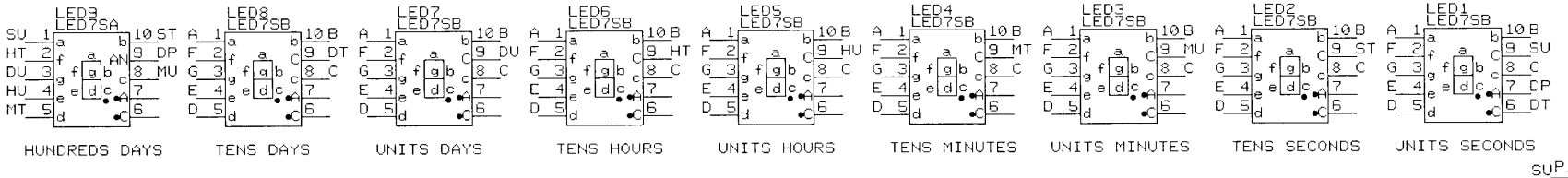
QTY	JXN	DESCRIPTION	COMMENTS	REF. DES.
1	0700002	CONNECTOR, 1 X 2 .025 SQ HEADER	N/U MASK	P22
2	0700003	CONNECTOR, 1 X 3 .025 SQ HEADER		P8,P13
6	0700003	CONNECTOR, 1 X 3 .025 SQ HEADER	N/U	P18,P20,P21,P24,P32, P34
1	0700005	CONNECTOR, 1 X 5 .025 SQ HEADER		P23
2	0700008	CONNECTOR, 1 X 8 .025 SQ HEADER	N/U MASK	P30,P31
1	0700010	CONNECTOR, 1 X 10 .025 SQ HEADER		P9
1	0700102	CONNECTOR, 2 X 2 .025 SQ HEADER	N/U MASK	P29
1	0700103	CONNECTOR, 2 X 3 .025 SQ HEADER		P6
1	0700105	CONNECTOR, 2 X 5 .025 SQ HEADER	N/U	P19
4	0700108	CONNECTOR, 2 X 8 .025 SQ HEADER		P3,P4,P5,P11
1	0700432	CONNECTOR, 1 X 32 .025 SQ FEMALE HEADER	!	P7
1	0700504	CONNECTOR, 2 X 4 FEMALE HEADER	OPT MASK	J6
1	0700532	CONNECTOR, 2 X 32 .025 SQ FEMALE HEADER	N/U MASK	P12
1	0700533	CONNECTOR, 2 X 33 .025 SQ FEMALE HEADER	!	P17
2	0701209	CONNECTOR, 1 X 9 2MM PITCH RT ANGLE HEADER	N/U MASK	P15,P16
1	0701504	CONNECTOR, 2 X 4 2MM SQ FEMALE HEADER	OPT MASK	J7
1	0701926	CONNECTOR, 26 PIN R/A FEMALE HI DENS DSUB		J3
1	0710028	SOCKET, 28 PIN PLCC		XU23
1	0710032	SOCKET, 32 PIN PLCC		XU20
2	0710044	SOCKET, 44 PIN PLCC		XU21,XU24
1	0710068	SOCKET, 68 PIN PLCC		XU11
1	0710084	SOCKET, 84 PIN PLCC		XU18
1	0799002	CONNECTOR, BNC JACK RT ANGLE LO PROFILE PC	*	J1
1	0799002	CONNECTOR, BNC JACK RT ANGLE LO PROFILE PC	OPT.	J2
2	0799004	CONNECTOR, 96 PIN EURO C		P1,P2
1	1010000	IC, Z84C00XXVEC Z80 CPU -40/+85C 6,8,10MHZ PLCC4	1010003 OK	U21
1	1010001	IC, Z84C44XXVEC Z80 SIO 6,8,10MHZ -40/+85C	-GM OPT.	U24
1	1010002	IC, 82C54 - TRI PROG CTR CMOS PLCC28		U23
1	1030026	IC, OP27GS LOW NOISE PRECISION OP AMP	OPT.	U36
1	1090001	IC, A1240XL FPGA, VME32D 84 PLCC		U18
1	1090006	IC, EPROM 32KX8 150NS -40/+85C PLCC32		U20
1	1090025	IC, A40MX02 FPGA, VME32I 68 PLCC		U11
2	1203330	RESISTOR, SURF MT. 3.3K OHM 1/10W 5% CF	* -xOUT U31-	RMOD1,RMOD2
1	1321001	CAPACITOR, 100PF 5% NPO MLC	!LED1-A/C	CMOD1
1	1321001	CAPACITOR, 100PF 5% NPO MLC	!LED2-A/C	CMOD2
2	1351001	CAPACITOR, .1UF Z5U MLC	OPT.	C24,C25
1	1700505	CONNECTOR, 2 X 5 .025 SQ FEMALE HEADER (SM)	GM OPT	J5
4	300-024	74ALS245 - ALS TTL OCTAL BUS TRANSCEIVER		U4,U5,U7,U8
2	4000012	HOLDER T1 LED 90 DEG		LEDHOLD1,LEDHOLD2
1	4000013	SPACER .230 (UNDER LEDHOLD1)		SP2301
1	4000013	SPACER .230 (UNDER LEDHOLD2)		SP2302
1	4000014	SPACER .350 (BETWEEN LED1 & LEDHOLD1)		SP3501
1	4000014	SPACER .350 (BETWEEN LED2 & LEDHOLD2)		SP3502
1	4000015	LED FILTER RED .6 IN X 3.2 IN.		LEDFIL1
1	400001B	FRONT PANEL VME-SYNCCLOCK32 REV B		FP1
7	4900000	LABEL .9 X .25		LABU9,LABU10,LABU11 LABU16,LABU17,LABU18 LABU31
1	5000305	PWB VME-SYNCCLOCK32 REV F		PWB1
1	9999901	9 DIGIT LED DISPLAY	LED9 OPT	LED9

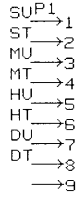
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

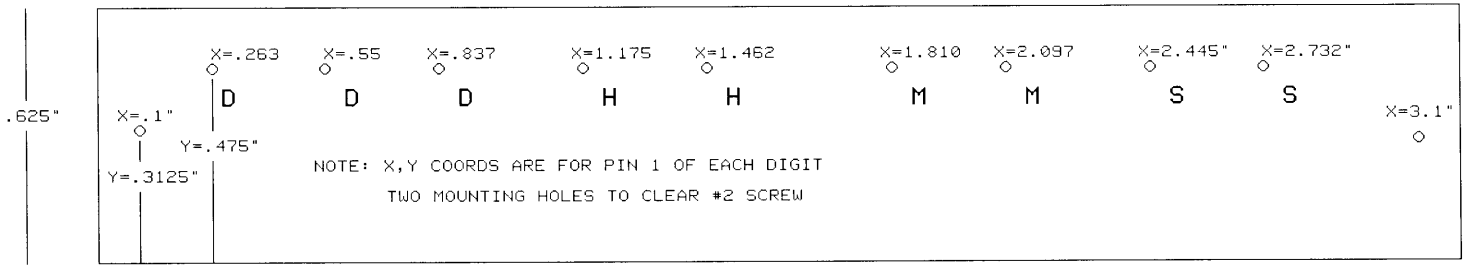


LED DIGIT DIMENSIONS (TOP VIEW)









100S OF DAYS SEGMENTS A..G
DRIVEN BY DP FOR LED1..LED7

LED1 DP DRIVEN BY DP FOR LED8

5000400
PWB1

15DEC95 JCK PWB1 JSN -> 5000400

JXI2, INC.
2308 LAKE OMEGA, EL PASO, TX 79936, USA

9 DIGIT LED DISPLAY

D		A0.P01	REV A0
DATE: 15DEC95 JCK			SHEET 01 OF 1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

BILL OF MATERIALS 9 DIGIT LED DISPLAY Fri Dec 15 09:08:34 1995 1

QTY	KS2N	DESCRIPTION	COMMENTS	REF. DES.
8	0120001	LED, 7 SEG+D.P. COMMON CATHODE		LED1,LED2,LED3,LED4 LED5,LED6,LED7,LED8
1	0120002	LED, 7 SEG+D.P. COMMON ANODE		LED9
1	0286047	RESISTOR NETWORK, 47 OHM BUSSED 10 PIN SIP		RP2
2	0701209	CONNECTOR, 1 X 9 RT ANG 2MM .031" HEADER		P1,P2
1	5000400	PWB 9 DIG LED REV A		PWB1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
                                ;*****************************************************************
                                ;   ****** users may need to modify this file or make their own *******
                                ;       VMESYN32.DEF 23SEP96 JCK FIX Ext_Polarity definition.
                                ;       VMESYN32.DEF 23APR96 JCK DEFINES HW ADDRESSES FOR VME-SYNCCLOCK32
                                ;       When assembled, produces VMESYN32.o68 for linking with application
                                ;       programs to define hardware addresses.
                                ;       The following defs apply to a Motorola MVME-133 host using A16
                                ;       access to a VME-SYNCCLOCK32 configured for a A16 base address
                                ;       of 0000. The MVME-133 uses different high order address bits
                                ;       for A16/D32 accesses (FEFF) than for A16/D08(o) accesses (FFFF).
                                ;       Users must consult their host processors hardware manual to
                                ;       determine the high order address bits, and then modify this
                                ;       file or generate one of their own per their hardware requirements.
                                ;*****************************************************************
                                        section         .vmebase,addr=$FFFF0001
ffff0001                        Heartbeat_Intr_Ctl:     ds.b    1       ; ffff0001  BIM Section A Ctrl
ffff0002                                                ds.b    1
ffff0003                        Match_Intr_Ctl:         ds.b    1       ; ffff0003  BIM Section B Ctrl
ffff0004                                                ds.b    1
ffff0005                        Time_Tag_Intr_Ctl:      ds.b    1       ; ffff0005  BIM Section C Ctrl
ffff0006                                                ds.b    1
ffff0007                        RAM_Fifo_Ctl:           ds.b    1       ; ffff0007  BIM Section D Ctrl
ffff0008                                                ds.b    1
ffff0009                        Heartbeat_Intr_Vec:     ds.b    1       ; ffff0009  BIM Section A Vec
ffff000a                                                ds.b    1
ffff000b                        Match_Intr_Vec:         ds.b    1       ; ffff000b  BIM Section B Vec
ffff000c                                                ds.b    1
ffff000d                        Time_Tag_Intr_Vec:      ds.b    1       ; ffff000d  BIM Section C Vec
ffff000e                                                ds.b    1
ffff000f                        Ram_Fifo_Vec:           ds.b    1       ; ffff000f  BIM Section D Vec
ffff0010                                                ds.b    1
ffff0011                        RAMFifo:                ds.b    1       ; ffff0011
ffff0012                                                ds.b    1
ffff0013                        Status:                 ds.b    1       ; ffff0013
ffff0014                                                ds.b    1
ffff0015                        Ext_100ns:              ds.b    1       ; ffff0015
ffff0016                                                ds.b    1
ffff0017                        Ext_Polarity:           ds.b    1       ; ffff0017
ffff0018                                                ds.b    1
ffff0019                                                ds.b    8       ; ffff0019,1b,1d,1f
ffff0021                        DPData:                 ds.b    1       ; ffff0021
ffff0022                                                ds.b    1
ffff0023                        DPAddr:                 ds.b    1       ; ffff0023
ffff0024                                                ds.b  1
ffff0025                        Resets:                 ds.b    1       ; ffff0025
ffff0026                                                ds.b    1
ffff0027                                                ds.b    1       ; ffff0027
ffff0028                                                ds.b    1
ffff0029                        Matchlo:                ds.b    1       ; ffff0029
ffff002a                                                ds.b    1
ffff002b                        Matchhi:                ds.b    1       ; ffff002b

                                        section         .vmelong,addr=$FEFF0018
feff0018                        Sec40_Usec1:    ds.b    4       ; 18
feff001c                        Year8_Min1:     ds.b    4       ; 1c
feff0020                                        ds.b    4       ; 20
feff0024                                        ds.b    4       ; 24
feff0028                        Ext_Sec40_Usec1: ds.b   4       ; 28
feff002c                        Ext_Year8_Min1: ds.b    4       ; 2c
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* VME32DEF.H 23APR96 JCK VME-SYNCCLOCK32 register bit definitions */
/* see also VMESYN32.DEF for defining register addresses */
/* Bit assignments for:
 Heartbeat_Intr_Ctl,Match_Intr_Ctl,Time_Tag_Intr_Ctl, Ram_Fifo_Intr_Ctl */
#define Int_Enb                 0x10    /* Interrupt enable for CH A-D */
#define Int_Autoclear           0x08    /* Interrupt autoclear for CH A-D */
#define IRQ_Level_1             0x01    /* IRQ level 1 pattern for CH A-D */
#define IRQ_Level_2             0x02    /* IRQ level 2 pattern for CH A-D */
#define IRQ_Level_3             0x03    /* IRQ level 3 pattern for CH A-D */
#define IRQ_Level_4             0x04    /* IRQ level 4 pattern for CH A-D */
#define IRQ_Level_5             0x05    /* IRQ level 5 pattern for CH A-D */
#define IRQ_Level_6             0x06    /* IRQ level 6 pattern for CH A-D */
#define IRQ_Level_7             0x07    /* IRQ level 7 pattern for CH A-D */
/* Bit assignments for Status */
#define Ext_Ready               1       /* Ext. Time Tag Data Ready if 1*/
#define Sync_OK                 2       /* In-sync to time reference if 1 */
#define RAM_FIFO_Ready          4       /* Ram_Fifo_Offset byte Ready if 1 */
#define Match                   8       /* Match register toggles on Match */
#define Heartbeat               0x10    /* Heartbeat pulse detected if 1 */
#define Tag16_Ready             0x40
#define Response_Ready          0x80    /* READ ONLY Command/dual port response ready */
/* Bit assignments for Resets */
#define Assert_Board_Reset      0xfe    /* WR bit0 = 0: Assert cold start */
#define Release_Board_Reset     0xfd    /* WR bit1 = 0: Release cold start */
#define Trigger_Sim_Ext_Time_Tag 0xfb    /* WR bit2 = 0: Fake ext. time tag */
#define Reset_Match             0xf7    /* WR bit3 = 0: Reset Match reg */
#define Reset_Heartbeat         0xef    /* WR bit4 = 0: Reset Heartbeat flag*/
#define Reset_Tag16             0xdf    /* WR bit5 = 0: Reset TAG16 FIFO */
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* VMEDPDEF>H 24MAR04 JCK add BATTIM_Holdoff and Jam_s10_s1, fix irig-g sel */
/* VMEDPDEF.H 22MAY99 JCK add extended match time definitions */
/* VMEDPDEF.H 29 JUL 97 JCK fix DP_CodeSelect_NASA36, add DP_CodeSelect_IRIGA*/
/* VMEDPDEF.H 11 JAN 97 JCK add GPS_Readlock,GPS_Readunlock*/
/* DVIDEF.H 19 NOV 96 JCK*/
#define DP_Command 0xff
#define No_op                    0 /* no operation */
#define Command_Set_Major        2 /* Set clock seconds..days to Major seconds..days */
#define Command_Set_Years        4 /* Set clock years to dual port years  */
#define Command_Set_RAM_FIFO     6 /* Set RAM FIFO external time tag mode */
#define Command_Reset_RAM_FIFO   8 /* Reset RAM FIFO external time tag mode */
#define Command_Empty_RAM_FIFO  10 /* Empty RAM FIFO */
#define Command_Set_Ctr0        12 /* Set 82C54 ctr 2 ("lowrate") params */
#define Command_Set_Ctr1        14 /* Set 82C54 ctr 2 ("heartbeat") params */
#define Command_Set_Ctr2        16 /* Set 82C54 ctr 2 ("rate2") params */
#define Command_Rejam           18 /* Re-jam at start of next second */
#define Command_Spfun1          20 /* Reserved for special function #1 */
#define Command_Spfun2          22 /* Reserved for special function #2 */
#define GPS_Readlock            24 /* lock GPS position buffer for read */
#define GPS_Readunlock          26 /* unlock GPS position buffer to allow update*/
/*   DUAL PORT RAM LOCATIONS 0x00..0x7F ARE READ-ONLY */
#define DP_Extd_Sts 0x00 /* Extended status READ ONLY */
#define DP_Extd_Sts_Nosync 0x01 /* Set if NOT in sync */
#define DP_Extd_Sts_Nocode 0x02 /* Set if selected input code NOT decodeable */
#define DP_Extd_Sts_NoPPS  0x04 /* Set if PPS input invalid */
#define DP_Extd_Sts_NoMajT 0x08 /* Set if major time NOT set since jam */
#define DP_Extd_Sts_NoYear 0x10 /* Set if year NOT set */
#define DP_Code_CtlA 0x0f       /* Control field "A" read data */
#define DP_Code_CtlB 0x10       /* Control field "B" read data */
#define DP_Code_CtlC 0x11       /* Control field "C" read data */
#define DP_Code_CtlD 0x12       /* Control field "D" read data */
#define DP_Code_CtlE 0x13       /* Control field "E" read data */
#define DP_Code_CtlF 0x14       /* Control field "F" read data */
#define DP_Code_CtlG 0x15       /* Control field "G" read data */


#define DP_Control0 0x80 /* Dual Port Ram Address for Control Register */
#define DP_Control0_Leapyear     0x01 /* Current year is leap year*/
#define DP_Control0_CodePriority 0x02 /* Time code input has priority over PPS*/
#define DP_Control0_NegCodePropD 0x04 /* code input prop delay setting is - */
#define DP_Control0_NegPPSPropD  0x08 /* PPS input prop delay setting is - */

#define DP_CodeSelect 0x81              /* Time code input select */
#define DP_CodeSelect_IRIGB 0x0b        /* IRIG-B */
#define DP_CodeSelect_IRIGA 0x0a        /* IRIG-A */
#define DP_CodeSelect_NASA36 0x06       /* NASA36 */
#define DP_CodeSelect_2137 0x07         /* 2137   */
#define DP_CodeSelect_XR3 0x03          /* XR3    */
//#define DP_CodeSelect_IRIGG 0x0f        /* IRIG-G */
#define DP_CodeSelect_IRIGG 0x09        /* IRIG-G */
#define DP_CodeSelect_IRIGE 0x0e        /* IRIG-E */

#define DP_LeapSec_Day10Day1 0x82       /* Day (10s & 1s) ending in leap sec*/

#define DP_LeapSec_Day1000Day100 0x83   /* Day (0,100s) ending in leap sec*/

#define DP_CodePropDel_ns100ns10  0x84  /* time code prop. delay 100,10 ns */
#define DP_CodePropDel_us10us1    0x85  /* time code prop. delay 10,1 us */
#define DP_CodePropDel_ms1us100   0x86  /* time code prop. delay 10,1 us */
#define DP_CodePropDel_ms100ms10  0x87  /* time code prop. delay 100,10 ms */


#define DP_PPS_PropDel_ns100ns10  0x88       /* PPS prop. delay 100,10 ns  */
#define DP_PPS_PropDel_us10us1    0x89       /* PPS prop. delay 10,1 us    */
#define DP_PPS_PropDel_ms1us100   0x8a       /* PPS prop. delay 1000,100 us*/
#define DP_PPS_PropDel_ms100ms10  0x8b       /* PPS prop. delay 100,10 ms  */

#define DP_PPS_Time_ns100ns10     0x8c          /* PPS time 100,10ns */
#define DP_PPS_Time_us10us1       0x8d          /* PPS time 10,1us */
#define DP_PPS_Time_ms1us100      0x8e          /* PPS time 1000,100us */
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
#define DP_PPS_Time_ms100ms10     0x8f          /* PPS time 100,10ms */
#define DP_Major_Time_s10s1       0x90          /* Major time 10,1second */
#define DP_Major_Time_m10m1       0x91          /* Major time 10,1minute */
#define DP_Major_Time_h10h1       0x92          /* Major time 10,1hour */
#define DP_Major_Time_d10d1       0x93          /* Major time 10,1 day */
#define DP_Major_Time_d1000d10    0x94           /* Major time 0, 100 day */
#define DP_Year10Year1        0x95              /* 10,1 years */
#define DP_Year1000Year100    0x96              /* 1000,100 years */
#define DP_Codebypass         0X97              /* #frames to validate code */
#define DP_Ctr2_ctl           0x98     /* ctr 2 control word */
#define DP_Ctr1_ctl           0x99     /* ctr 1 control word */
#define DP_Ctr0_ctl           0x9A     /* ctr 0 control word */
#define DP_Ctr2_ctl_sel                 0x80  /* ALWAYS used for ctr 2 */
#define DP_Ctr1_ctl_sel                 0x40  /* ALWAYS used for ctr 1 */
#define DP_Ctr0_ctl_sel                 0x00  /* ALWAYS used for ctr 0 */
#define DP_ctl_rw                       0x30  /* ALWAYS used */
#define DP_ctl_mode0                    0x00  /* Ctrx mode 0 select */
#define DP_ctl_mode1                    0x02  /* Ctrx mode 1 select */
#define DP_ctl_mode2                    0x04  /* Ctrx mode 2 select */
#define DP_ctl_mode3                    0x06  /* Ctrx mode 3 select */
#define DP_ctl_mode4                    0x08  /* Ctrx mode 4 select */
#define DP_ctl_mode5                    0x0A  /* Ctrx mode 5 select */
#define DP_ctl_bin                      0x00  /* Ctrx binary mode select */
#define DP_ctl_bcd                      0x01  /* Ctrx bcd mode select */
#define DP_Ctr2_lsb           0x9B     /* ctr 2 count LSB    */
#define DP_Ctr2_msb           0x9C     /* ctr 2 count MSB    */
#define DP_Ctr1_lsb           0x9D     /* ctr 1 count LSB    */
#define DP_Ctr1_msb           0x9E     /* ctr 1 count MSB    */
#define DP_Ctr0_lsb           0x9F     /* ctr 0 count LSB    */
#define DP_Ctr0_msb           0xA0     /* ctr 0 count MSB    */
#define DP_BATTIM_Holdoff_m10m1 0xBD  /* BATTIM wait (minutes) to set Major Time */
#define DP_BATTIM_Jam_s10s1   0xC4     /* BATTIM wait (seconds) to copy bat time to clock w/o 1pps */
#define DP_Match_ht           0xBE     /* -EXTMATCH option: match time .ht (BCD) */
#define DP_Match_SS           0xBF     /* -EXTMATCH option: match time ss (BCD)  */
#define DP_Match_MM           0xCB     /* -EXTMATCH option: match time mm (BCD)  */
#define DP_Match_HH           0xCC     /* -EXTMATCH option: match time hh (BCD)  */
#define DP_Match_D1D0         0xCD     /* -EXTMATCH option: match time day (10,1 digits) (BCD) */
#define DP_Match_XD2          0xCE     /* -EXTMATCH option: match time don't care, 100 days digit */
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* curtime.c 23apr96 demonstrates reading current time from VME-SYNCCLOCK32 */
/* also demonstrates reading extended status from dual port memory */
#include <stdio.h>
#include "vme32def.h"
#include "vmedpdef.h"
int Read_Dual_Port_RAM(char addr);
main()
{int i,timelo,timehi;
/* the following declaration is for the hardware time low (sec,usec) and
time hi (units of years throgh minutes) 32 bit registers. When linked with
a hardware definition file (for example: VMESYN32) the correct hardware
addresses will be filled in by the linker. An alternative approach would
be to use pointer variables initialized to the hardware addresses, but the
Introl C compiler doesn't allow that approach */
extern int Sec40_Usec1, Year8_Min1;
while (1)
  {
  /* read timelo first, latching both lo & hi-order word. Then read timehi. */
  timelo=Sec40_Usec1;
  timehi=Year8_Min1;
  i=Read_Dual_Port_RAM(DP_Extd_Sts);
  printf("%08x %08x Extd_status: %2x \n" ,timehi,timelo,i);
  }
return (0);
}

int Read_Dual_Port_RAM(char addr)
{
extern char DPData,DPAddr,Status;
char response;
response=DPData; /* clear response flag */
DPAddr=addr;
while ( (Status&Response_Ready) ==0) {}
response=DPData;
return response;
}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* extintr.c 23apr96 JCK demonstrates processing external time tag interrupts.
        ******** IMPORTANT: For this program to actually run, a source of time
        tag pulses must be connected to the External Time Tag input pin.
        Looping back the CLK0 output (J3 pin 7) to the External Time Tag
        input (J3 pin 3) works nicely for demonstration purposes. */
#include <stdio.h>
#include <introl.h>
#include "vme32def.h"
extern char Time_Tag_Intr_Ctl,Time_Tag_Intr_Vec;
int xhi,xlo;
main()
{
/* Introl C uses  __mod2__ type modifier to define a interrupt handler */
void   __mod2__ timetaghandler(void);
/* the Introl C _VECTOR function is used to set an entry in a interrupt
   vector table (entry 25 in this case) to point to the interrupt handler.
   Users of other compilers will need to use the interrupt support facilities
   of their compiler and run time libraries which will probably be different */
_VECTOR(timetaghandler,25);     /* set interrupt vector table entry */
Time_Tag_Intr_Vec=25;           /* set VME-SYNCCLOCK32 to same vector number */
Time_Tag_Intr_Ctl=Int_Enb+IRQ_Level_1;
while (!0) {printf (" %08x %08x \n",xhi,xlo);}
return (0);
}
/*  this is the actual interrupt handler code.
When the VME-SYNCCLOCK asserts the programmed IRQ signal
The Interrupt Controller (usually on the VME host processor) then in turn -
if the interrupt request level is >= processor mask level -
will try to interrupt the host processor. When the interrupt controller accepts the
interrupt the VME-SYNCCLOCK32 will supply an interrupt vector
(the value 25 written into Time_Tag_Intr_Vec in this example).
The VME processor will then use the interrupt vector to determine
the address of the interrupt routine. */

void   __mod2__ timetaghandler(void)
{   extern int Ext_Sec40_Usec1,Ext_Year8_Min1;
    xlo=Ext_Sec40_Usec1;
    xhi=Ext_Year8_Min1;
    /* background process continuously displays xhi,xlo */
}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* mattst.c 23apr96 JCK test match time function of VME-SYNCCLOCK32
Running time about 100 seconds for 9999 different match time settings.
**** LOOP back MAT output (J3-6) to EXTTAG input (J3-3) ****
By looping back MATCH output (MAT signal) as external time tag input (EXTTAG),
the program can check that not only did MAT toggle - but that it also toggled
at the correct time. */
#include <stdio.h>
#include "vme32def.h"
main()
{int matus3,matus2,matus1,matus0,match,xtime,xmsec,xusec;
 extern char Ext_Polarity,Matchhi,Matchlo,Status,Resets;  /* See VMESYN32.DEF */
 extern int Ext_Sec40_Usec1;                              /* See VMESYN32.DEF */
 Ext_Polarity=0;        /* use ext time tag ^ edge */
 for (match=9999;match!=9900;match=((match+99)%10000) )
  {
  matus3=match/1000;       /* get 1000 usec BCD digit */
  matus2=(match%1000)/100; /* get 100 usec BCD digit */
  matus1=(match%100)/10;   /* get 10 usec BCD digit */
  matus0=(match%10);       /* get 1 usec BCD digit */
  Resets=Reset_Match;      /* Reset Match Status */
  Matchhi=(matus3*16)+matus2; /* set new BCD match time */
  Matchlo=(matus1*16)+matus0;
  while (!(Status&Match)){;} /* wait for time match  */
  Matchhi=0xff;             /* disable match toggle /
  Matchlo=0xff;             /* disable match toggle */
  xtime=Ext_Sec40_Usec1 & 0x0000ffff ;
  xmsec=(xtime&0x00000f000)/0x1000;
  xusec=(((xtime&0x000000f00)/0x100)*100)
       +(((xtime&0x0000000f0)/0x10)*10)
       +(xtime&0x00000000f);
    if ((xusec != (match %1000)) || (xmsec!= (match/1000)))
     {printf(".%03ld%03ld %04d \n " ,xmsec,xusec,match);}
}
 printf (" completed ");
return (0);
}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* heart.c 16apr97 JCK set hb rate to 125 PPS */
#include <stdio.h>
#include <introl.h>
#include "vme32def.h"
#include "vmedpdef.h"
extern char Heartbeat_Intr_Ctl,Heartbeat_Intr_Vec,Resets;
extern int Sec40_Usec1,Year8_Min1;
int thi,tlo,hbcount;
int Set_Dual_Port_RAM(char addr,char value);
main()
{
/* Introl C uses  __mod2__ type modifier to define a interrupt handler */
void   __mod2__ timetaghandler(void);
/* the Introl C _VECTOR function is used to set an entry in a interrupt
   vector table (entry 25 in this case) to point to the interrupt handler.
   Users of other compilers will need to use the interrupt support facilities
   of their compiler and run time libraries which will probably be different */
_VECTOR(timetaghandler,25);     /* set interrupt vector table entry */
Heartbeat_Intr_Vec=25;           /* set VME-SYNCCLOCK32 to same vector number */
Heartbeat_Intr_Ctl=Int_Enb+IRQ_Level_1;
 Set_Dual_Port_RAM (DP_Ctr1_lsb, 0xc0); /* 125=  3000000/24000  */
 Set_Dual_Port_RAM (DP_Ctr1_msb, 0x5d); /* 24000 = hex 5dc0 */
 Set_Dual_Port_RAM (DP_Command,Command_Rejam);  /* align to 1pps */
 hbcount=0;
 while (1)   /* display time now */
  {
  if (hbcount >= 125)   /* display every 125th heartbeat intr time */
        {hbcount=0;
        printf("%d %8x %8x\n ",hbcount,thi,tlo);
        }
  }
}
int Set_Dual_Port_RAM(char addr, char value)
{
extern char DPData,DPAddr,Status;
char response;
int wait;
for (wait=0;wait<=50;wait++)
     {response=DPData;} /* clear response flag & delay*/
DPAddr=addr;
while ((Status&Response_Ready) ==0) {}
for (wait=0;wait<=50;wait++)
     {response=DPData;} /* discard old RAM contents & delay*/
DPData=value;    /* set new contents */
while ((Status&Response_Ready) ==0) {}
response=DPData;
return response;
}

/*  this is the actual interrupt handler code.
When the VME-SYNCCLOCK asserts the programmed IRQ signal
The Interrupt Controller (usually on the VME host processor) then in turn -
if the interrupt request level is >= processor mask level -
will try to interrupt the host processor. When the interrupt controller accepts the
interrupt the VME-SYNCCLOCK32 will supply an interrupt vector
(the value 25 written into Time_Tag_Intr_Vec in this example).
The VME processor will then use the interrupt vector to determine
the address of the interrupt routine. */

void   __mod2__ timetaghandler(void)
{   extern int _Sec40_Usec1,Year8_Min1;
    tlo=Sec40_Usec1;
    thi=Year8_Min1;
    hbcount++;
    Resets=Reset_Heartbeat;     /* clear heartbeat ready flag */
}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* setmajor.c 23apr96 JCK convert from PC-104 regs to VME-SYNCCLOCK32 */
/* setmajor.c 15mar96 JCK use dual port RAM to set major time */
#include <stdio.h>
#include "vme32def.h"
#include "vmedpdef.h"
int Set_Dual_Port_RAM(char addr,char value);
main()
{extern int Sec40_Usec1,Year8_Min1;
 int thi,tlo,days,hours,minutes,secs,msec;
 Set_Dual_Port_RAM (DP_Major_Time_d1000d10, 0x03);
 Set_Dual_Port_RAM (DP_Major_Time_d10d1,0x65);
 Set_Dual_Port_RAM (DP_Major_Time_h10h1,0x23);
 Set_Dual_Port_RAM (DP_Major_Time_m10m1,0x59);
 Set_Dual_Port_RAM (DP_Major_Time_s10s1,0x55);
 Set_Dual_Port_RAM (DP_Command,Command_Set_Major);  /* Set 365 23:59:55 */
 while (1)   /* display time now */
  {
  tlo=Sec40_Usec1;       /* LATCH ALL DIGITS NOW */
  thi=Year8_Min1;
  days=(((thi&0x0f000000)/0x01000000)*100)+(((thi&0x00f00000)/0x00100000)*10)
       +((thi&0x000f0000)/0x00010000);
  hours=  (((thi&0x0000f000)/0x00001000)*10)+(thi&0x00000f00)/0x00000100;
  minutes=(((thi&0x000000f0)/0x00000010)*10)+(thi&0x0000000f) ;
  secs=   (((tlo&0xf0000000)/0x10000000)*10)+(tlo&0x0f000000)/0x01000000;
  msec=  (((tlo&0x00f00000)/0x00100000)*100)+(((tlo&0x000f0000)/0x00010000)*10)
          +((tlo&0x00000f00)/0x00000100);
  printf(" %03d %02d:%02d:%02d.%03d\n " ,days,hours,minutes,secs,msec);
  }
}
int Set_Dual_Port_RAM(char addr, char value)
{
extern char DPData,DPAddr,Status;
char response;
response=DPData; /* clear response flag */
DPAddr=addr;
while ((Status&Response_Ready) ==0) {}
response=DPData; /* read & discard old RAM contents */
DPData=value;    /* set new contents */
while ((Status&Response_Ready) ==0) {}
response=DPData;
return response;
}
int Set_Dual_Port_RAM(char addr, char value)
{
extern char DPData,DPAddr,Status;
char response;
response=DPData; /* clear response flag */
DPAddr=addr;
while ((Status&Response_Ready) ==0) {}
response=DPData; /* read & discard old RAM contents */
DPData=value;    /* set new contents */
while ((Status&Response_Ready) ==0) {}
response=DPData;
return response;
}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* ramfifo.c 23apr96 JCK convert from PC-104 to VME
/* ramfifo.c 15mar96 JCK use RAM FIFO buffer for external time tags */
#include <stdio.h>
#include "vme32def.h"
#include "vmedpdef.h"
int Set_Dual_Port_RAM(char addr,char value);
extern char RAMFifo,DPAddr,DPData,Status;
main()
{int timetag[9],i,evtcnt=0;
 Set_Dual_Port_RAM (DP_Command,Command_Empty_RAM_FIFO); /* empty RAM FIFO */
 timetag[0]=RAMFifo;                  /* discard old FIFO out */
 Set_Dual_Port_RAM (DP_Command,Command_Set_RAM_FIFO); /* set RAM FIFO tags */
 while (1)
 {
  for (i=0;i<=9;i++)
  {while ((Status & RAM_FIFO_Ready)==0) {;} /* wait Ready*/
   timetag[i]=RAMFifo&0x000000ff;
  }
  evtcnt=(evtcnt+1)%10; /* display every 10th time tag */
  if (evtcnt==0) {printf ("%02x%02x %02x:%02x:%02x.%01x%02x%02x%02x %02x \n",
        timetag[9],timetag[8],timetag[7],timetag[6],timetag[5],
        timetag[4]/16,timetag[3],timetag[2],timetag[1],timetag[0]);}
 }
}
int Set_Dual_Port_RAM(char addr, char value)
{
char response;
response=DPData; /* clear response flag */
DPAddr=addr;
while ((Status&Response_Ready) ==0) {}
response=DPData;
DPData=value;
while ((Status&Response_Ready) ==0) {}
response=DPData;
return response;
}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* gpsdat.c 11jan97 JCK read & display GPS lat/lon/nav status from VME-SYNC32*/
/* requires -GR, -GT, or -GM option for meaningful data. Else shows all 0s. */
#include <stdio.h>
#include "vme32def.h"
#include "vmedpdef.h"
#include "gpsdef.h"
int Read_Dual_Port_RAM(char addr);
int Set_Dual_Port_RAM(char addr,char value);
extern char RAMFifo,DPAddr,DPData,Status;
main()
{int latbcddegw,latbcdminw,latbcdhiw,latbcdlow,status,readlock_response,i,j;
 int lonbcddeg32w,lonbcddegw,lonbcdminw,lonbcdhiw,lonbcdlow,lockcnt=0;
 char EW,NS,NA;
 long latdeg,londeg,latmin,lonmin,latfrachi,latfraclo,lonfrachi,lonfraclo;
while (1)
     {
        readlock_response=Set_Dual_Port_RAM (DP_Command,GPS_Readlock); /* attempt read lock */
        for (i=0;i<=100;i++) j=Status; /* delay > 50 usec or so between commands  */
        if   (readlock_response < 0) lockcnt++;
        else
        {
        status=Read_Dual_Port_RAM(DP_GPS_Status);
        latbcddegw=Read_Dual_Port_RAM(DP_GPS_Lat_DegE10);
        latbcdminw=Read_Dual_Port_RAM(DP_GPS_Lat_MinE10);
        latbcdhiw=Read_Dual_Port_RAM(DP_GPS_Lat_MinEM12);
        latbcdlow=Read_Dual_Port_RAM(DP_GPS_Lat_MinEM34);
        lonbcddeg32w=Read_Dual_Port_RAM(DP_GPS_Lon_DegE32);
        lonbcddegw=Read_Dual_Port_RAM(DP_GPS_Lon_DegE10);
        lonbcdminw=Read_Dual_Port_RAM(DP_GPS_Lon_MinE10);
        lonbcdhiw=Read_Dual_Port_RAM(DP_GPS_Lon_MinEM12);
        lonbcdlow=Read_Dual_Port_RAM(DP_GPS_Lon_MinEM34);
        /* unlock after nav data collected to allow update */
        Set_Dual_Port_RAM (DP_Command,GPS_Readunlock);
        latdeg=(((latbcddegw&0xf0)/16)*10)+(latbcddegw&0x0f);
        latmin=(((latbcdminw&0xf0)/16)*10)+(latbcdminw&0x0f);
        latfrachi=(((latbcdhiw&0xf0)/16)*10)+(latbcdhiw&0x0f);
        latfraclo=(((latbcdlow&0xf0)/16)*10)+(latbcdlow&0x0f);
        londeg=((lonbcddeg32w&0x0f)*100)+(((lonbcddegw&0xf0)/16)*10)+(lonbcddegw&0x0f);
        lonmin=(((lonbcdminw&0xf0)/16)*10)+(lonbcdminw&0x0f);
        lonfrachi=(((lonbcdhiw&0xf0)/16)*10)+(lonbcdhiw&0x0f);
        lonfraclo=(((lonbcdlow&0xf0)/16)*10)+(lonbcdlow&0x0f);
        if (status&DP_GPS_Status_West)  EW='W'; else EW='E';
        if (status&DP_GPS_Status_South) NS='S'; else NS='N';
        if (status&DP_GPS_Status_Nav)   NA='N'; else NA='A';
        printf("%c sats:%d %c%02ld %02ld.%02ld%02ld  %c%03ld %02ld.%02ld%02ld %d\n ",
         NA,status&0x0f,NS,latdeg,latmin,latfrachi,latfraclo,
         EW,londeg,lonmin,lonfrachi,lonfraclo,lockcnt);
        }
   }
}
int Set_Dual_Port_RAM(char addr, char value)
{
char response;
response=DPData; /* clear response flag */
DPAddr=addr;
while ((Status&Response_Ready) ==0) {}
response=DPData;
DPData=value;
while ((Status&Response_Ready) ==0) {}
response=DPData;
return response;
}

int Read_Dual_Port_RAM(char addr)
{
extern char DPData,DPAddr,Status;
char response; int delay,j;
response=DPData; /* clear response flag */
DPAddr=addr;
while ( (Status&Response_Ready) ==0) {}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
response=DPData;
for (delay=0;delay<100;delay++) j=Status; /* 100 VME bus accesses about 40 usec */
return response;
}
```

```
/* gpsbin.c 12aug00 JCK display DOP type,rcvr sts,sawtooth too VME-SYNC32 */
/* gpsbin.c 31jul97 JCK read & display binary GPS lat/lon/nav status from VME-SYNC32 */
/* requires -GM option for meaningful data. Else shows all 0s. */
/* also demonstrates using update flag to detect position updates */
#include <stdio.h>
#include "vme32def.h"
#include "vmedpdef.h"
#include "gpsdef.h"
int Read_Dual_Port_RAM(char addr);
int Set_Dual_Port_RAM(char addr,char value);
extern char RAMFifo,DPAddr,DPData,Status;
main()
{int latdeg,latsec,latmin,latmsec,londeg,lonmin,lonsec,lonmsec,status,i,j,rtcmack;
 int DOP_Type,RcvrStatus,SawtoothErr;
 char EW,NS,NA;
 long latbin,lonbin;
 while (1)
     {  while (Read_Dual_Port_RAM(DP_GPS_UpdateFlag)==0)
              for (i=0;i<=100;i++) j=Status; /* delay > 50 usec  */
        for (i=0;i<=100;i++) j=Status;       /* delay > 50 usec  */
        Set_Dual_Port_RAM(DP_GPS_UpdateFlag, 0); /* clear for next time */
        status=Read_Dual_Port_RAM(DP_GPS_Status);
        rtcmack=Read_Dual_Port_RAM(DP_GPS_RTCM_Ack)&0x00ff;
        DOP_Type=Read_Dual_Port_RAM(DP_GPS_DOP_Type);
        RcvrStatus=Read_Dual_Port_RAM(DP_GPS_RcvrStatus);
        SawtoothErr=Read_Dual_Port_RAM(DP_GPS_SawtoothErr);
        latbin=Read_Dual_Port_RAM(DP_GPS_Latbin0700)&0x000000ff;
        latbin=latbin+((Read_Dual_Port_RAM(DP_GPS_Latbin1508)&0x000000ff)<<8);
        latbin=latbin+((Read_Dual_Port_RAM(DP_GPS_Latbin2316)&0x000000ff)<<16);
        latbin=latbin+((Read_Dual_Port_RAM(DP_GPS_Latbin3124)&0x000000ff)<<24);
        if (latbin<0)
            {NS='S';
             latbin=-latbin;
            }
        else NS='N';
        latdeg=latbin/(60*60*1000);
        latmin=latbin%(60*60*1000)/(60*1000);
        latsec=(latbin/1000)%60;
        latmsec=latbin%1000;
        lonbin=Read_Dual_Port_RAM(DP_GPS_Lonbin0700)&0x000000ff;
        lonbin=lonbin+((Read_Dual_Port_RAM(DP_GPS_Lonbin1508)&0x000000ff)<<8);
        lonbin=lonbin+((Read_Dual_Port_RAM(DP_GPS_Lonbin2316)&0x000000ff)<<16);
        lonbin=lonbin+((Read_Dual_Port_RAM(DP_GPS_Lonbin3124)&0x000000ff)<<24);
        if (lonbin<0)
            {EW='W';
             lonbin=-lonbin;
            }
        else EW='E';
        londeg=lonbin/(60*60*1000);
        lonmin=lonbin%(60*60*1000)/(60*1000);
        lonsec=(lonbin/1000)%60;
        lonmsec=lonbin%1000;
        if (status&DP_GPS_Status_Nav)   NA='N'; else NA='A';
        printf("%c sats:%d RTCM Ack:%03d %c%02d %02d'%02d.%03d\" %c%03d %02d'%02d.%03d DOPT:%02x
Rsts:%02x Saw:%02d \n",
         NA,status&0x0f,rtcmack,NS,latdeg,latmin,latsec,latmsec,
         EW,londeg,lonmin,lonsec,lonmsec,
         DOP_Type,RcvrStatus,SawtoothErr);
     }
}
int Set_Dual_Port_RAM(char addr, char value)
{
char response;int delay,j;
response=DPData; /* clear response flag */
DPAddr=addr;
while ((Status&Response_Ready) ==0) {}
response=DPData;
for (delay=0;delay<100;delay++) j=Status; /* 100 VME bus accesses about 40 usec */
DPData=value;
while ((Status&Response_Ready) ==0) {}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
response=DPData;
for (delay=0;delay<100;delay++) j=Status; /* 100 VME bus accesses about 40 usec */
return response;
}

int Read_Dual_Port_RAM(char addr)
{
extern char DPData,DPAddr,Status;
char response; int delay,j;
response=DPData; /* clear response flag */
DPAddr=addr;
while ( (Status&Response_Ready) ==0) {}
response=DPData;
for (delay=0;delay<100;delay++) j=Status; /* 100 VME bus accesses about 40 usec */
return response;
}
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP