# STEbus Interface

# for Acorn RISC OS-based Computer Systems

# User Guide and Programmer's Reference Manual

Intelligent Interfaces Ltd

30th September 1996

All information is given by Intelligent Interfaces in good faith. However, it is acknowledged that there may be errors or omissions in this User Guide. Intelligent Interfaces welcomes comments and suggestions relating to the STEbus System for Acorn RISC OS-based Computer Systems and this User Guide.

All correspondence should be addressed to:-

Technical Enquiries
P O Box 80
Eastleigh
Hants
SO53 2YX

Tel: 023 8026 1514
Fax: 087 0052 128
Email: andy@intint.demon.co.uk

This User Guide is intended only to assist the reader in the use of the STEbus System for Acorn RISC OS-based Computer Systems and, therefore, Intelligent Interfaces shall not be liable for any loss or damage whatsoever arising from the use of any information or particulars in, or any error or omission in, this User Guide, or any in-correct use of the STEbus System for Acorn RISC OS-based Computer Systems.

# Contents


## 1 Introduction

1.1 Fitting the Computer Expansion Card

## 2 The STEbus Interface Hardware

2.1 Hardware Overview

2.2 Hardware Details

## 3 The STEbus Interface Software

3.1 The STEbus Module SWI's

## 4 STEbus Interface Link Selection Options

4.1 Computer Expansion Card

4.2 STEbus Board

## Appendices

I Memory Map
II Extended Expansion Card ID
III STEbus Module SWI's

## Circuit Diagram

# 1 Introduction

The STEbus Interface Expansion Card fitted in any Acorn Risc OS-based Computer System with an expansion backplane enables the use of an external STEbus input/output sub-system for measurement and control applications. The large range of STEbus I/O boards available include Analogue to Digital and Digital to Analogue converters, Parallel I/O (including counter/timers), Serial I/O, Networking Interfaces (Bit Bus, HART, etc.), Stepper and Servo Motor Controllers, EPROM Programmers, etc.

The STEbus (IEEE Std 1000) Interface comprises two single Eurocards connected by up to 2m of ribbon cable. One card plugs into the expansion backplane of the computer and the other plugs into the STEbus backplane.

The interface enables the computer to directly address either the 4 Kbyte STEbus I/O space or the 1 Mbyte STEbus memory space (in 256 x 4 Kbyte pages).

The card which plugs into the backplane of the STEbus system provides STEbus system controller facilities. It is a potential master of the STEbus and has an on-card arbiter which enables the computer to be used in high performance multi-master systems.

Full software support is provided by a module.

## 1.1 Fitting the Expansion Card

The card can be fitted in any Acorn Computer with an expansion backplane.

To fit the card in an A300 series, A400 series, A540, A5000, Risc PC or A7000 computer:-

1 Switch off the power to the Computer.
2 Disconnect the Computer from the mains supply.
3 The card can be fitted in any unused expansion card slot.
4 Remove the blanking plate from the rear of the Computer and retain the two screws.
5 Fit the card and secure it in position using the two screws retained at stage 4 (if required, fit a joiner and blanking plate).
6 Reconnect the Computer to the mains supply.
7 Switch on the power to the Computer.
8 Confirm that the card has been fitted correctly by pressing F12 and typing

```
*Podules
```

This should list the STEbus as

```
Intelligent Interfaces STEbus Interface
```

together with any other cards fitted. Press <Return> to return to the Desktop.

The maximum number of STEbus Expansion Cards that can be fitted is limited only by the number of unused expansion card slots.

# 2 The STEbus Interface Hardware

## 2.1 Hardware Overview

### System Controller Facilities

The STEbus Interface provides the following facilities:-

i) SYSCLK, a 16 MHz system clock signal. This is a link selectable option.

ii) SYSRST*, a bit in the control latch on the STEbus board enables the system reset signal to be under the control of software running on the computer. This is a link selectable option. Note that the STEbus system reset SYSRST* signal and the reset RST* signal of the computer are not directly connected allowing the two systems to be reset independently.

### On-board Arbitration

The STEbus board provides an arbiter which, as supplied, implements an arbitration algorithm which initially allocates bus control to requests in the following order of priority - local RQ*, BUSRQ0*, BUSRQ1* and then works on a 'round robin' basis to ensure that no request is ignored forever. Other arbitration algorithms can be supplied to special order.

### On-board/Off-board Arbitration

The on-board arbiter is a link selectable option and can be disabled. The local RQ is then linked to either BUSRQ0 or BUSRQ1 to request bus control from an arbiter on another STEbus board.

### Master Type and Mode

The STEbus Interface is a potential master which, as supplied, works on a 'Release-On-Request' basis through the detection of another Bus Request BUSRQ* signal becoming active. Alternatively it can, if supplied to special order, work on a 'Release-When-Done' basis. The 'Release-On-Request' mode reduces the number of bus arbitrations required and thereby reduces the average computer STEbus access time. When the computer, via the STEbus Interface, is the only master in an STEbus system, the first arbitration is the only one ever carried out.

A bit in the control latch on the STEbus board enables a request by the STEbus Interface for control of the bus to be locked. This allows the ARM processor of the computer to perform the indivisible read modify write operations essential in a multi-master system.

### Addressing

A bit in the control latch on the STEbus board enables the selection of either the 4 Kbyte of the STEbus I/O space or the STEbus memory space. A paging register is provided on the STEbus board to enable the 1 Mbyte of STEbus memory space to be addressed as 256 x 4 Kbyte pages. The 4 Kbyte of STEbus address space is mapped directly into the 4 Kbyte x 8 of MEMC podule address space allocated to each expansion card.

A state machine on the expansion card synchronises the MEMC cycle with the asynchronous handshake protocol of the STEbus. This, in turn, is implemented by a state machine on the STEbus board which involves the synchronisation lines address strobe ADRSTB*, data strobe DATSTB*, data transfer acknowledge DATACK* and transfer error TFRERR*.

### Interrupts

There are two main sources of interrupts.

i) STEbus Attention Requests ATNRQ*<0...7) interrupts. These can be individually linked to

generate either an IRQ (ordinary) or FIQ (fast) interrupt to the ARM processor of the computer. A control bit in the control latch on the STEbus board allows the Attention Request interrupts to be enabled or disabled as a group. An Attention Request status register is provided to enable the source of the interrupt to be determined.

ii) Cycle Error CYCERR* and System Reset SYSRST* interrupts. A Cycle Error CYCERR* interrupt is generated if the ARM processor of the computer attempts to access the STEbus and a System Reset SYSRST* is in progress or a Transfer Error TFRERR* occurs. A System Reset SYSRST* interrupt is generated whenever System Reset SYSRST* is active whether it is being driven by the STEbus interface or another board in the system.

The Cycle Error CYCERR* and System Reset SYSRST* interrupts can be individually linked to generate either IRQ or FIQ interrupts to the ARM processor of the computer. A control bit in the control latch on the STEbus board allows the two interrupts to be enabled or disabled as a group. A status register is provided to enable the source of the interrupt, CYCERR* or SYSRST*, to be determined.

The two interrupts are reset by a write to a reserved address.

Two control bits in the control latch on the expansion card allow all IRQ and FIQ interrupts from the STEbus interface to be independently enabled and disabled. A status register is provided to enable the source of the interrupt, IRQ or FIQ to be determined.

### Bus driver and receivers

The bus drivers and receivers on the STEbus board meet the specification given in the IEEE Std 1000-1987.

## 2.2 Hardware Details

### STEbus I/O or memory space

Computer addresses
Slot 0 - &03000000 to &03003FFC
Slot 1 - &03004000 to &03007FFC
Slot 2 - &03008000 to &0300BFFC
Slot 3 - &0300C000 to &03003FFC
(4 Kbyte x 8 of MEMC podule address space - each byte on a 32 bit word boundary)

Computer cycle - MEMC Read/Write

### Computer expansion card status register

Computer addresses
Slot 0 - &03342800
Slot 1 - &03346800
Slot 2 - &0334A800
Slot 3 - &0334E800

Computer cycle - Fast Read

| | |
|---|---|
| 0 | IRQ interrupt status (0 = no IRQ interrupt, 1 = IRQ interrupt) |
| 1 | 0 |
| 2 | FIQ interrupt status (0 = no FIQ interrupt, 1 = FIQ interrupt) |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | reflects the state of latch[6] IRQ interrupt enable (0 = IRQ disabled, 1 = IRQ enabled) |
| 7 | reflects the state of latch[7] FIQ interrupt enable (0 = FIQ disabled, 1 = FIQ enabled) |

### Computer expansion card control latch

Computer addresses
Slot 0 - &03342800
Slot 1 - &03346800
Slot 2 - &0334A800
Slot 3 - &0334E800

Computer cycle - Fast Write
0 A10
1 A11
2 A12
3 A13
4 A14
5 A16
6 IRQ interrupt enable (0 = IRQ disabled, 1 = IRQ enabled)
7 FIQ interrupt enable (0 = FIQ disabled, 1 = FIQ enabled)

### STEbus board status register

Computer addresses
Slot 0 - &03343000
Slot 1 - &03347000
Slot 2 - &0334B000
Slot 3 - &0334F000

Computer cycle - Fast Read
0 reflects the state of latch[0] ATN interrupt enable
(0 = disabled, 1 = enabled)
1 reflects the state of latch[1] CYCERR and SYSRST interrupt enable
(0 = disabled, 1 = enabled)
2 reflects the state of latch[2] RESET (0 = false, 1 = true)
3 reflects the state of latch[3] MEM I/O*
(0 = STEbus I/O address space selected,
1 = STEbus memory address space selected)
4 reflects the state of latch[4] LOCK
(0 = requests for control of the STEbus not locked,
1 = requests for control of the STEbus locked)
5 CYCERR interrupt status (0 = no CYCERR interrupt, 1 = CYCERR interrupt)
6 SYSRST interrupt status (0 = no SYSRST interrupt, 1 = SYSRST interrupt)
7 SYSRST status (0 = SYSRST* false, 1 = SYSRST* true)

## STEbus board control latch

Computer addresses
Slot 0 - &03343000
Slot 1 - &03347000
Slot 2 - &0334B000
Slot 3 - &0334F000

Computer cycle - Fast Write
0 ATN interrupt enable (0 = disabled, 1 = enabled)
1 CYCERR and SYSRST interrupt enable (0 = disabled, 1 = enabled)
2 RESET if link S7 fitted (0 = SYSRST* false, 1 = SYSRST* true)
3 MEM I/O* (0 = select STEbus I/O address space,
1 = select STEbus memory address space)
4 LOCK (0 = requests for control of the STEbus not locked,
1 = requests for control of the STEbus locked)
5 unused state not relevant
6 unused state not relevant
7 unused state not relevant

### STEbus board clear CYCERR and SYSRST interrupts

Computer addresses
Slot 0 - &03343004
Slot 1 - &03347004
Slot 2 - &0334B004
Slot 3 - &0334F004

Computer cycle - Fast Write
0 unused state not relevant (write to reserved addres clears interrupts)
1 unused state not relevant (write to reserved addres clears interrupts)
2 unused state not relevant (write to reserved addres clears interrupts)
3 unused state not relevant (write to reserved addres clears interrupts)
4 unused state not relevant (write to reserved addres clears interrupts)
5 unused state not relevant (write to reserved addres clears interrupts)
6 unused state not relevant (write to reserved addres clears interrupts)
7 unused state not relevant (write to reserved addres clears interrupts)

### STEbus board Attention Request status register

Computer addresses
Slot 0 - &03343008
Slot 1 - &03347008
Slot 2 - &0334B008
Slot 3 - &0334F008

Computer cycle - Fast Read
0 ATNRQ0* status (0 = no ATNRQ0, 1 = ATNRQ0)
1 ATNRQ0* status (0 = no ATNRQ1, 1 = ATNRQ1)
2 ATNRQ0* status (0 = no ATNRQ2, 1 = ATNRQ2)
3 ATNRQ0* status (0 = no ATNRQ3, 1 = ATNRQ3)
4 ATNRQ0* status (0 = no ATNRQ4, 1 = ATNRQ4)
5 ATNRQ0* status (0 = no ATNRQ5, 1 = ATNRQ5)
6 ATNRQ0* status (0 = no ATNRQ6, 1 = ATNRQ6)
7 ATNRQ0* status (0 = no ATNRQ7, 1 = ATNRQ7)

The Attention Request status register inverts the state of the ATNRQ* lines

### STEbus board address latch

Computer addresses
Slot 0 - &0334300C
Slot 1 - &0334700C
Slot 2 - &0334B00C
Slot 3 - &0334F00C

Computer cycle - Fast Write
0 A12
1 A13
2 A14
3 A15
4 A16
5 A17
6 A18
7 A19

The address latch forms an STEbus memory space 4Kbyte page register

# 3 The STEbus Interface Software

The STEbus Interface software is a relocatable module which extends the RISC OS operating system. The module is stored in ROM on the expansion card and is automatically loaded and initialised by RISC OS when the computer is switched on or reset. The module provides I/O boards used with the computer. It also provides a highly optimised interrupt handling routine for interrupts from the STEbus system.

The STEbus Interface software provides SWI's to enable any of the ten sources of IRQ interrupts (ATNRQ*<0...7>, CYCERR* and SYSRST*) to be claimed and released in a similar manner to the SWI's OS_Claim Device Vector and OS_releaseDevice Vector provided by the RISC OS operating system. This simplifies the writing of interrupt routines.

## 3.1 The STEbus Module SWI's

### STEbus_rdARCsts (SWI &420C0)

**Purpose**
To read the status register on the expansion card.

**Parameters**
R0 - expansion card slot number

**Results**
R2 - status
status[0] - IRQ interrupt status (0 = no IRQ interrupt, 1 = IRQ interrupt)
status[2] - FIQ interrupt status (0 = no FIQ interrupt, 1 = FIQ interrupt)
status[6] - reflects the state of latch[6] IRQ interrupt enable
(0 = IRQ disabled, 1 = IRQ enabled)
status[7] - reflects the state of latch[7] FIQ interrupt enable
(0 = FIQ disabled, 1 = FIQ enabled)

**Example**

```
10 computer_slot% = 1
20 SYS "STEbus_rdARCsts",computer_slot% TO ,,status%
```

returns the contents of the status register in the variable status%.

### STEbus_wrARClat (SWI &420C1)

**Purpose**
To write to the control latch on the expansion card.

**Parameters**
R0 - expansion card slot number
R2 - latch
latch[5:0] - ROM page register
latch[6] - IRQ interrupt enable (0 = IRQ disabled, 1 = IRQ enabled)
latch[7] - FIQ interrupt enable (0 = FIQ disabled, 1 = FIQ enabled)

**Example**

```
10 computer_slot% = 3
20 SYS "STEbus_wrARClat",computer_slot%,,%01000000
```

writes %01000000 to the control latch to enable IRQ interrupts.

### STEbus_rdSTEsts (SWI &420C2)

**Purpose**
To read the status register on the STEbus board.

**Parameters**
R0 - expansion card slot number

**Results**
R2 - status
status[0] - reflects the state of latch[0] ATN interrupt enable
(0 = disabled, 1 = enabled)
status[1] - reflects the state of latch[1] CYCERR and SYSRST interrupt enable
(0 = disabled, 1 = enabled)
status[2] - reflects the state of latch[2] RESET (0 = false, 1 = true)
status[3] - reflects the state of latch[3] MEM/IO*
(0 = STEbus I/O address space selected,
1 = STEbus memory address space selected)
status[4] - reflects the state of latch[4] LOCK
(0 = requests for control of the STEbus not locked,
1 = requests for control of the STEbus locked)
status[5] - CYCERR interrupt status
(0 = no CYCERR interrupt, 1 = CYCERR interrupt)
status[6] - SYSRST interrupt status
(0 = no SYSRST interrupt, 1 = SYSRST interrupt)
status[7] - SYSRST status (0 = SYSRST false, 1 = SYSRST true)

**Example**

```
10 computer_slot% = 1
20 SYS "STEbus_rdSTEsts",computer_slot% TO ,,status%
```

returns the contents of the status register in the variable status%.

### STEbus_wrSTElat (SWI &420C3)

**Purpose**
To write to the control latch on the STEbus board.

**Parameters**
R0 - expansion card slot number
R2 - latch
latch[0] - ATN interrupt enable (0 = disabled, 1 = enabled)
latch[1] - CYCERR and SYSRST interrupt enable
(0 = disabled, 1 = enabled)
latch[2] - RESET if link S7 is fitted
(0 = SYSRST false, 1 = SYSRST true)
latch[3] - MEM/IO* (0 = select STEbus I/O address space
1 = select STEbus memory address space
latch[4] - LOCK (0 = requests for control of the STEbus not locked,
1 = requests for control of the STEbus locked

**Example**

```
10 computer_slot% = 0
20 SYS "STEbus_wrSTElat",computer_slot%,,%00000001
```

writes %00000001 to the control latch to enable ATN interrupts.

### STEbus_enableIRQ (SWI &420C4)

Pur**p**ose
To set the IRQ interrupt enable bit[6] in the control latch on the expansion card.

**Parameters**
R0 - expansion card slot number

**Example**

```
10 computer_slot% = 3
20 SYS "STEbus_enableIRQ",computer_slot%
```

enables IRQ interrupts from the expansion card in slot 3.

### STEbus_disableIRQ (SWI &420C5)

**Purpose**
To clear the IRQ interrupt enable bit[6] in the control latch on the expansion card.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 3
20 SYS "STEbus_disableIRQ",computer_slot%
```

disables IRQ interrupts from the expansion card in slot 3.

### STEbus_enableFIQ (SWI &420C6)

**Purpose**
To set the FIQ interrupt enable bit[7] in the control latch on the expansion card.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_enableFIQ",computer_slot
```

enables FIQ interrupts from the expansion card in slot 1.

### STEbus_disableFIQ (SWI &420C7)

**Purpose**
To clear the FIQ interrupt enable bit[7] in the control latch on the expansion card.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_disableFIQ",computer_slot%
```

disables FIQ interrupts from the expansion card in slot 1.

### STEbus_enableATNinterrupts (SWI &420C8)

**Purpose**
To set the ATN interrupt enable bit[0] in the control latch on the STEbus board.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_enableATNinterrupts",computer_slot%
```

enables ATN interrupts from the STEbus interface in slot 1.

### STEbus_disableATNinterrupts (SWI &420C9)

**Purpose**
To clear the ATN interrupt enable bit[0] in the control latch on the STEbus board.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_disableATNinterrupts",computer_slot%
```

disables ATN interrupts from the STEbus interface in slot 1.

### STEbus_enableSYSRSTandCYCERRinterrupts (SWI &420CA)

**Purpose**
To set the CYCERR and SYSRST interrupt enable bit[1] in the control latch on the STEbus board.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 2
20 SYS "STEbus_enableSYSRSTand CYCERRinterrupts",computer_slot%
```

enables CYCERR and SYSRST interrupts from the STEbus interface in slot 2.

### STEbus_disableSYSRSTandCYCERRinterrupts (SWI &420CB)

**Purpose**
To clear the CYCERR and SYSRST interrupt enable bit[1] in the control latch on the STEbus board.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 2
20 SYS "STEbus_disableSYSRSTandCYCERRinterrupts",computer_slot%
```

disables CYCERR and SYSRST interrupts from the STEbus interface in slot 2.

### STEbus_setSYSRST (SWI &420CC)

**Purpose**
To set the RESET bit[2] in the control latch on the STEbus board if link S7 is fitted SYSRST is set true.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_setSYSRST",computer_slot%
```

if link S7 is fitted on the STEbus interface in slot 1 SYSRST is set true.

### STEbus_clearSYSRST (SWI &420CD)

**Purpose**
To clear the RESET bit[2] in the control latch on the STEbus board if link S7 is fitted SYSRST is set false.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_clearSYSRST",computer_slot%
```

if link S7 is fitted on the STEbus interface in slot 1 SYSRST is set false.

### STEbus_selectMEM (SWI &420CE)

**Purpose**
To set the MEM/IO* bit[3] in the control latch on the STEbus board to select STEbus memory address space.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_selectMEM",computer_slot%
```

selects memory address space for the STEbus interface in slot 1.

### STEbus_selectIO (SWI &420CF)

**Purpose**
To clear the MEM/IO* bit[3] in the control latch on the STEbus board to select STEbus I/O address space.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_selectIO",computer_slot%
```

selects I/O address space for the STEbus interface in slot 1.

### STEbus_lock (SWI &420D0)

**Purpose**
To set the LOCK bit[4] in the control latch on the STEbus board to lock requests for control of the STEbus. This enables the computer to perform indivisible read modify write operations essential in a multi-master system.

**Parameters**
R0 - expansion card slot number

**Example**
```
10 computer_slot% = 1
20 SYS "STEbus_selectMEM",computer_slot%
```

locks requests for control of the STEbus connected via the STEbus interface in slot 1.

### STEbus_unlock (SWI &420D1)

**Purpose**
To clear the LOCK bit[4] in the control latch on the STEbus board.

**Parameters**
R0 - expansion card slot number

**Example**

```
10 computer_slot% = 1
20 SYS "STEbus_selectMEM",computer_slot%
```

unlocks requests for control of the STEbus connected via the STEbus interface in slot 1.

### STEbus_reset (SWI &420D2)

**Purpose**
To set the reset bit in the control latch on the STEbus board for 350m S. If link S7 is fitted (as supplied) this resets the STEbus.

**Parameters**
R0 - expansion card slot number

**Example**

```
10 computer_slot% = 2
20 SYS "STEbus_reset"
```

### STEbus_STEclri (SWI &420D3)

**Purpose**
To clear Cycle Error CYCERR and System Reset SYSRST interrupts.

**Parameters**
R0 - expansion card slot number

**Example**

```
10 computer_slot% = 1
20 SYS "STEbus_STEclri",computer_slot%
```

clears Cycle Error CYCERR and System Reset SYSRST interrupts from the STEbus interface in slot 1.

### STEbus_rdATN (SWI &420D4)

**Purpose**
To read the Attention Request status register on the STEbus board.

**Parameters**
R0 - expansion card slot number

Results
R2 - status
status[0] ATNRQ0* status (0 = no ATNRQ0, 1 = ATNRQ0)
status[7] ATNRQ7* status (0 = no ATNRQ7, 1 = ATNRQ7)

### STEbus_wrADR (SWI &420D5)

**Purpose**
To write to the address latch on the STEbus board (STEbus memory space 4 Kbyte page register).

**Parameters**
R0 - expansion card slot number R2 - page (0-255)

**Example**

```
10 computer_slot% = 1
20 page% = &0A
30 SYS "STEbus_wrADR",computer_slot%,,page%
```

selects STEbus memory space 4 Kbyte page &0A (address range &0A000-&0AFFF) for the STEbus interface in slot 1.

### STEbus_rdIO (SWI &420D6)

**Purpose**
To read a byte from STEbus IO address space.

**Parameters**
R0 - expansion card slot number R1 - STE IO address

**Results**
R2 - data

**Example**

```
10 computer_slot% = 0
20 SPINCadr% = &3C0
30 SYS "STEbus_rdIO",computer_slot%,SPINCadr%+2 TO ,,data%
```

reads Port A of the Z8536 CIO1 (address at Base+2) on an Arcom SPINC board and returns the value in variable data%.

### STEbus_wrIO (SWI &420D7)

**Purpose**
To write a byte to STEbus IO address space.

**Parameters**
R0 - expansion card slot number R1 - STE IO address R2 - data

**Example**

```
10 computer_slot% = 0
20 SPINCadr% = &3C0
30 data% = %11000100
40 SYS "STEbus_wrIO",computer_slot%,SPINCadr%+2,data%
```

writes the bit pattern 11000100 to Port A of the Z8536 CIO1 (address at Base+2) on an Arcom SPINC board.

### STEbus_rdMEM (SWI &420D8)

**Purpose**
To read a byte from STEbus memory address space.

**Parameters**
R0 - expansion card slot number R1 - STE memory address

Results
R2 - data

**Example**

```
10 computer_slot% = 1
20 memadr% = &0C040
```

```
30 SYS "STEbus_rdMEM",computer_slot%,memadr% TO ,,data%
```

reads the STEbus memory at address &0C040 and returns the value in variable data%.

### STEbus_wrMEM (SWI &420D9)

**Purpose**
To write a byte to STEbus memory address space.

**Parameters**
R0 - expansion card slot number
R1 - STE memory address
R2 - data

**Example**

```
10 computer_slot% = 1
20 memadr% = &3A173
30 data% = &A5
40 SYS "STEbus_wrMEM",computer_slot%,memadr%,data%
```

writes the byte &A5 to the STEbus memory at address &3A173.

### STEbus_rdMEMblk (SWI &420DA)

**Purpose**
To read a block of bytes from STEbus memory address space into the computer's memory.

**Parameters**
R0 - expansion card slot number
R1 - start STE memory address
R2 - start computer memory address
R3 - number of bytes to read

**Example**

```
10 computer_slot% = 2
20 STEmemadr% = &00C00
30 DIM computer_memadr% 1024
40 count% = 1024
50 SYS
"STEbus_rdMEMblk",computer_slot%,STEmemadr%,computer_memadr%,count%
```

reads 1024 bytes from STEbus memory starting at address &00C00 into the computer's memory reserved by the DIM statement (start address defined by the variable computer_memadr%).

### STEbus_wrMEMblk (SWI &420DB)

**Purpose**
To write a block of bytes to STEbus memory address space from the computer's memory.

**Parameters**
R0 - expansion card slot number
R1 - start STE memory address
R2 - start of computer's memory address
R3 - number of bytes to write

**Example**

```
10 computer_slot% = 1
20 STEmemadr% = &A0000
30 DIM computer_memadr% 2048
40 count% = 2048
```

```
.
.                          initialise the computer's memory
.

90 SYS
```

writes 2048 bytes to STEbus memory starting at address &A0000 from the computer's memory reserved by the DIM statement (start address defined by the variable computer_memadr%).

### STEbus_claim IRQvector (SWI &420DC)

**Purpose**
To claim an STEbus IRQ vector.

**Parameters**
R0 - expansion card slot number
R1 - interrupt source (0-9)
(0 - ATNRQ0*)
(7 - ATNRQ7*)
(8 - CYCERR*)
(9 - SYSRST*)
R2 - address of interrupt service routine
R3 - value to be passed in R12 when interrupt service routine called

### STEbus_releaseIRQvector (SWI &420DD)

**Purpose**
To release an STEbus IRQ vector.

**Parameters**
R0 - expansion card slot number
R1 - interrupt source (0-9)
(0 - ATNRQ0*)
(7 - ATNRQ7*)
(8 - CYCERR*)
(9 - SYSRST*)
R2 - address of interrupt service routine
R3 - value to be passed in R12 when interrupt service routine called

### STEbus_claim FIQ (SWI &420DE)

**Purpose**
To claim the FIQ interrupt, move the FIQ handler to &0000001C, change to FIQ mode and initialise FIQ registers, return to SVC mode and set the FIQ interrupt enable bit in the control latch on the expansion card.

**Parameters**
R0 - expansion card slot number
R1 - address of FIQ handler
R2-R5 - values to be passed in FIQ Regs R10-R13

### STEbus_releaseFIQ (SWI &420DF)

**Purpose**
Clear the FIQ interrupt enable bit in the control latch on the expansion card and release the FIQ interrupt.

**Parameters**
R0 - expansion card slot number

### STEbus_adrhw (SWI &420E0)

**Purpose**
To return the hardware addresses.

**Parameters**
R0 - expansion card slot number

**Results**
R0 - base address in the computer's memory of the STEbus I/O or memory space
(4 Kbyte x 8 of MEMC podule address space
- each byte on a 32 bit word boundary)
R1 - address of status register (read) and control latch (write) on the computer expansion card
R2 - address of status register (read) and control latch (write) on the STEbus board
R3 - address of location which when written to clears CYCERR and SYSRST interrupts
R4 - address of Attention Request status register on the STEbus board
R5 - address of Address latch on the STEbus board

### STEbus_adrmemADR (SWI &420E1)

**Purpose**
To return the address of the memory copy of the STEbus memory space page.

**Parameters**
R0 - expansion card slot number

**Results**
R1 - address of the memory copy of the STEbus memory space page

### STEbus_checkpresent (SWI &420E2)

**Purpose**
To determine whether an STEbus interface is present.

**Parameters**
R0 - expansion card slot number

**Results**
R1 - >0 expansion card present, 0 expansion card not present

### STEbus_rdmodwr IO (SWI &420E3)

**Purpose**
To perform an indivisible read modify write of I/O address space, requests for control of the bus are locked during the execution of this SWI. The new value = (old value AND R3) EOR R2.

**Parameters**
R0 - expansion card slot number
R1 - STE IO address
R2 - EOR mask
R3 - AND mask

**Example**

```
10 computer_slot% = 0
20 IOadr% = &3C0
30 EORmsk% = %10110000
40 ANDmsk% = %00001111
50 SYS "STEbus_rdmodwr IO",computer_slot%,EORmsk%,ANDmsk%
```

the new value of IO address &3C0 = (old value AND %00001111) EOR %10110000.

### STEbus_rdmodwrMEM (SWI &420E4)

**Purpose**
To perform an indivisible read modify write of memory address space, requests for control of the bus are locked during the execution of this SWI. The new value = (old value AND R3) EOR R2.

**Parameters**
R0 - expansion card slot number
R1 - STE memory address
R2 - EOR mask
R3 - AND mask

**Example**

```
10 computer_slot% = 0
20 memadr% = &3A000
30 EORmsk% = %10000000
40 ANDmsk% = %01111111
50 SYS "STEbus_rdmodwrMEM",computer_slot%,memadr%,EORmsk%,ANDmsk%
```

the new value of memory address &3C0 = (old value AND %00001111) EOR %10110000.

# 4 STEbus Interface Link Selection Options

Note - the following link descriptions are true when the link is fitted (+ indicates the link position as supplied).

## 4.1 Computer Expansion Card

| | EPROM Size | Link S1 | Link S2 | Link S3 |
|---|---|---|---|---|
| | 16K | A | A | A |
| + | 32K | B | A | A |
| | 64K | B | B | A |
| | 128K | B | B | B |

## 4.2 STEbus BOARD

### Bus request

| | | | |
|---|---|---|---|
| + | S1 | A | local request to on board arbiter |
| | S1 | B | request on BUSRQ0* |
| | S1 | C | request on BUSRQ1 |

### Bus acknowledge

| | | | |
|---|---|---|---|
| + | S2 | A | local acknowledge from on board arbiter |
| | S2 | B | acknowledge from BUSAK0* |
| | S2 | C | acknowledge from BUSAK1* |

Master Mode (Release on request enable)

| | | | |
|---|---|---|---|
| + | S3 | A | release on BUSRQ0 |
| + | S3 | B | release on BUSRQ1 |

On board arbiter connections

| | | | |
|---|---|---|---|
| + | S4 | A | connect arbiter to BUSAK0 |
| + | S4 | B | connect arbiter to BUSAK1 |

### Link Selection Options S1 S2 S3 and S4 Summary

### On board arbitration

### Bus request - local request (as supplied)

| | | | |
|---|---|---|---|
| S1 A fitted | S2 A fitted | S3 A fitted | S4 A fitted |
| S1 B open | S2 B open | S3 B fitted | S4 B fitted |
| S1 C open | S2 C open | | |

### Bus request on BUSRQ0*

| | | | |
|---|---|---|---|
| S1 A open | S2 A open | S3 A open | S4 A fitted |
| S1 B fitted | S2 B fitted | S3 B fitted | S4 B fitted |
| S1 C open | S2 C open | | |

### Bus request on BUSRQ1*

| | | | |
|---|---|---|---|
| S1 A open | S2 A open | S3 A fitted | S4 A fitted |
| S1 B open | S2 B open | S3 B open | S4 B fitted |
| S1 C fitted | S2 C fitted | | |

### External arbitration

### Bus request on BUSRQ0*

| | | | |
|---|---|---|---|
| S1 A open | S2 A open | S3 A open | S4 A open |
| S1 B fitted | S2 B fitted | S3 B fitted | S4 B open |
| S1 C open | S2 C open | | |

**Bus request on BUSRQ1***

| | | | |
|---|---|---|---|
| S1 A open | S2 A open | S3 A fitted | S4 A open |
| S1 B open | S2 B open | S3 B open | S4 B open |
| S1 C fitted | S2 C fitted | | |

**Interrupt enables**

| | | |
|---|---|---|
| S5 1A | ATNRQ0* | interrupts connected to the computer's IRQ |
| S5 1B | ATNRQ0* | interrupts connected to the computer's FIQ |
| S5 2A | ATNRQ1* | interrupts connected to the computer's IRQ |
| S5 2B | ATNRQ1* | interrupts connected to the computer's FIQ |
| S5 3A | ATNRQ2* | interrupts connected to the computer's IRQ |
| S5 3B | ATNRQ2* | interrupts connected to the computer's FIQ |
| S5 4A | ATNRQ3* | interrupts connected to the computer's IRQ |
| S5 4B | ATNRQ3* | interrupts connected to the computer's FIQ |
| S5 5A | ATNRQ4* | interrupts connected to the computer's IRQ |
| S5 5B | ATNRQ4* | interrupts connected to the computer's FIQ |
| S5 6A | ATNRQ5* | interrupts connected to the computer's IRQ |
| S5 6B | ATNRQ5* | interrupts connected to the computer's FIQ |
| S5 7A | ATNRQ6* | interrupts connected to the computer's IRQ |
| S5 7B | ATNRQ6* | interrupts connected to the computer's FIQ |
| S5 8A | ATNRQ7* | interrupts connected to the computer's IRQ |
| S5 8B | ATNRQ7* | interrupts connected to the computer's FIQ |
| S5 9A | CYCERR* | (Cycle Error) connected to the computer's IRQ |
| S5 9B | CYCERR* | (Cycle Error) connected to the computer's FIQ |
| S5 10A | SYSRST* | interrupts connected to the computer's IRQ |
| S5 10B | SYSRST* | interrupts connected to the computer's FIQ |

**SYSCLK select**

| | | |
|---|---|---|
| + | S6 | SYSCLK from the STEbus board (connects the STEbus SYSCLK bus signal to the 16 MHz signal on the STEbus board) |

**SYSRST select**

| | | |
|---|---|---|
| + | S7 | SYSRST* from the STEbus board (connects the STEbus SYSRST* bus signal to the open collector buffered output from the control latch on the STEbus board) |

# APPENDIX I

## MEMORY MAP

| | R/W | slot 0 | slot 1 | slot 2 | slot 3 |
|---|---|---|---|---|---|
| STEbus board address latch | W | &0334300C | &0334700C | &0334B00C | &0334F00C |
| STEbus board Attention Request status register | R | &03343008 | &03347008 | &0334B008 | &0334F008 |
| STEbus board clear CYCERR and SYSRST interrupts | W | &03343004 | &03347004 | &0334B004 | &0334F004 |
| STEbus board status register/latch control | R/W | &03343000 | &03347000 | &0334B000 | &0334F000 |
| Computer expansion card status register/latch control | R/W | &03342800 | &03346800 | &0334A800 | &0334E800 |
| Computer expansion card ID | R | &033C0000 | &033C4000 | &033C8000 | &033CC000 |
| STEbus I/O or memory space | R/W from to | &03000000 &03003FFC | &03004000 &03007FFC | &03008000 &0300BFFC | &0300C000 &0300FFFC |

# Appendix II

## Extended Expansion Card ID

Product Type (bytes 3-lo and 4-hi) = 118 STEbus Interface

Manufacturer Code (bytes 5-lo and 6-hi) = 5 Intelligent Interfaces

Country (byte 7) = 0 UK

# Appendix III

**STEbus Module SWI's (parameters normal, results italic)**

| | SWI | R0 | R1 | R2 | R3 | R4 | R5 | R6 |
|---|---|---|---|---|---|---|---|---|
| STEbus_rdARCsts | &420C0 | slot | | data | | | | |
| STEbus_wrARClat | &420C1 | slot | | data | | | | |
| STEbus_rdSTEsts | &420C2 | slot | | data | | | | |
| STEbus_wrSTElat | &420C3 | slot | | data | | | | |
| STEbus_enableIRQ | &420C4 | slot | | | | | | |
| STEbus_disableIRQ | &420C5 | slot | | | | | | |
| STEbus_enableFIQ | &420C6 | slot | | | | | | |
| STEbus_disableFIQ | &420C7 | slot | | | | | | |
| STEbus_enableATNinterrupts | &420C8 | slot | | | | | | |
| STEbus_disableATNinterrupts | &420C9 | slot | | | | | | |
| STEbus_enableSYSRSTandCYCERRinterrupts | &420CA | slot | | | | | | |
| STEbus_disableSYSRSTandCYCERRinterrupts | &420CB | slot | | | | | | |
| STEbus_setSYSRST | &420CC | slot | | | | | | |
| STEbus_clearSYSRST | &420CD | slot | | | | | | |
| STEbus_selectMEM | &420CE | slot | | | | | | |
| STEbus_selectIO | &420CF | slot | | | | | | |
| STEbus_lock | &420D0 | slot | | | | | | |
| STEbus_unlock | &420D1 | slot | | | | | | |
| STEbus_reset | &420D2 | slot | | | | | | |
| STEbus_STEclri | &420D3 | slot | | | | | | |
| STEbus_rdATN | &420D4 | slot | | data | | | | |
| STEbus_wrADR | &420D5 | slot | | data | | | | |
| STEbus_rdIO | &420D6 | slot | STEadr | data | | | | |
| STEbus_wrIO | &420D7 | slot | STEadr | data | | | | |
| STEbus_rdMEM | &420D8 | slot | STEadr | data | | | | |
| STEbus_wrMEM | &420D8 | slot | STEadr | data | | | | |
| STEbus_rdMEMblk | &420DA | slot | STEadr | compadr | count | | | |
| STEbus_wrMEMblk | &420DB | slot | STEadr | compadr | count | | | |
| STEbus_claimIRQvector | &420DC | slot | intsrc | rtadr | R12 | | | |
| STEbus_releaseIRQvector | &420DD | slot | intsrc | rtadr | R12 | | | |
| STEbus_claimFIQ | &420DE | slot | intsrc | rtadr | R10 | R11 | R12 | R13 |
| STEbus_releaseFIQ | &420DF | slot | | | | | | |
| STEbus_adrhw | &420E0 | slot | | | | | | |

Results (R1 MEMCadr, R2 STEstslatadr, R3 STEstslatadr, R4 STEclriadr, R5 ATNadr ADRadr)

| | SWI | R0 | R1 | R2 | R3 | R4 | R5 | R6 |
|---|---|---|---|---|---|---|---|---|
| STEbus_adrmemADR | &420E1 | slot | memADRadr | | | | | |
| STEbus_checkpresent | &420E2 | slot | | | | | | |
| | | status | | | | | | |
| STEbus_rdmodwrIO | &420E3 | slot | STEadr | EORmsk | ANDmsk | | | |
| STEbus_rdmodwrMEM | &420E4 | slot | STEadr | EORmsk | ANDmsk | | | |

**Notes**

| | |
|---|---|
| slot | expansion card slot number |
| data | refer to the description in the appropriate part of section 3 |
| STEadr | STE memory or IO address |
| compadr | computer's memory address |
| count | number of bytes to read or write |
| intsrc | interrupt source |
| rtadr | address of interrupt handler |
| R10,R11,R12,R13 | values to be passed in Regs R10-R13 |
| MEMCadr | base address in the computer's memory of the STEbus I/O or memory space |
| ARCstslatadr | address of status register (read) and control latch (write) on the computer expansion card |
| STEstslatadr | address of status register (read) and control latch (write) on the STEbus board |
| STEclriadr | address of location which when written to clears CYCERR and SYSRST interrupts |
| ATNadr | address of Attention Request status register on the STEbus board |
| ADRadr | address of Address latch on the STEbus board |
| memADRadr | address of the memory copy of the STEbus memory space page status<br>>0 expansion card present, 0 expansion card not present |
| EORmsk | EOR mask |
| ANDmsk | AND mask |

PL1
CONNECTOR DIN41612 TYPE C 90 DEG PLUG
ROWS A AND C LOADED

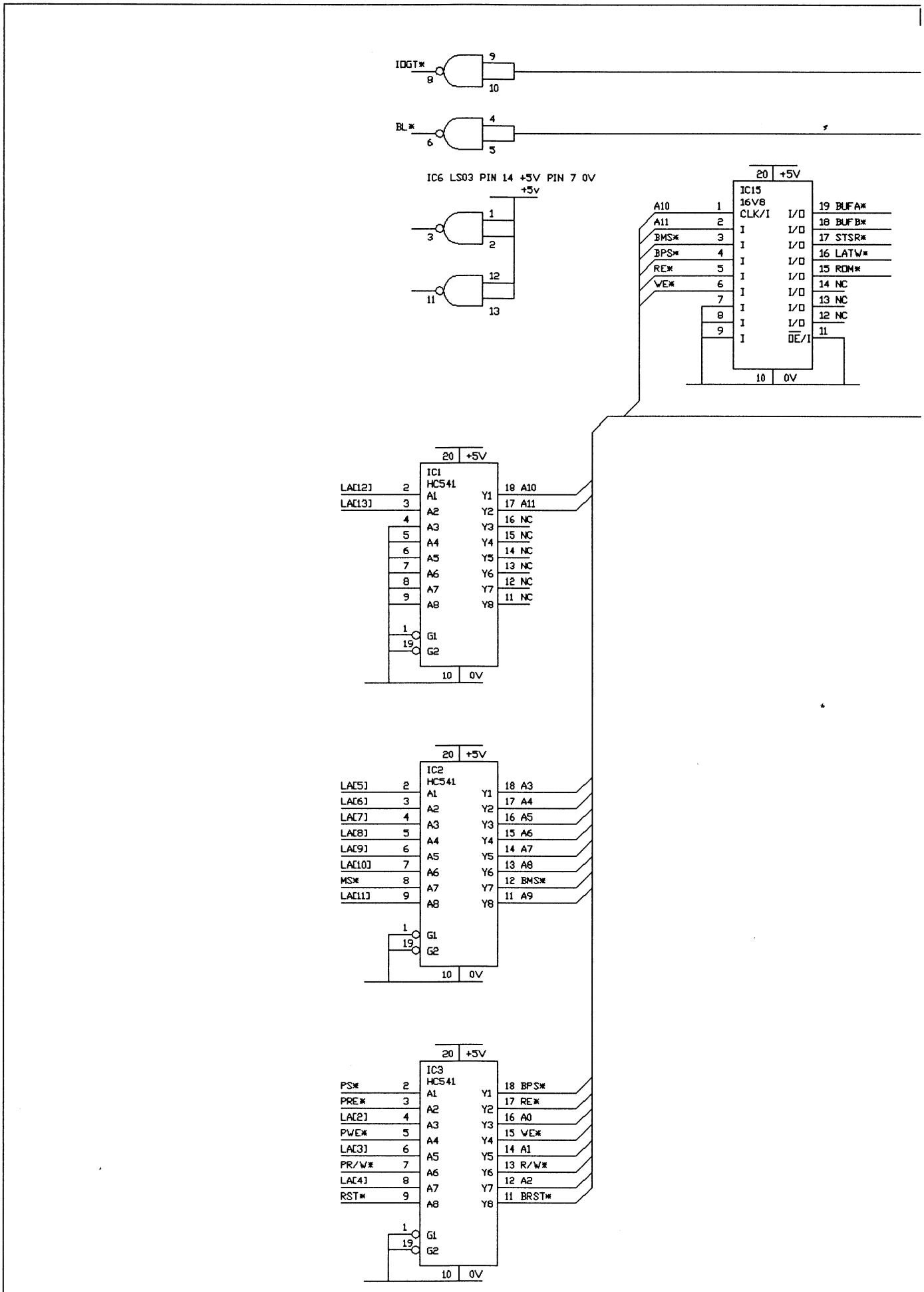
| | | | |
|---|---|---|---|
| 0V | A1 | C1 | 0V |
| NC | A2 | C2 | NC |
| NC | A3 | C3 | 0V |
| LA[13] | A4 | C4 | 0V |
| LA[12] | A5 | C5 | NC |
| LA[11] | A6 | C6 | MS* |
| LA[10] | A7 | C7 | NC |
| LA[9] | A8 | C8 | NC |
| LA[8] | A9 | C9 | NC |
| LA[7] | A10 | C10 | NC |
| LA[6] | A11 | C11 | NC |
| LA[5] | A12 | C12 | RST* |
| LA[4] | A13 | C13 | PR/W* |
| LA[3] | A14 | C14 | PWE* |
| LA[2] | A15 | C15 | PRE* |
| NC | A16 | C16 | PIRQ* |
| NC | A17 | C17 | PFIQ* |
| NC | A18 | C18 | NC |
| NC | A19 | C19 | NC |
| NC | A20 | C20 | NC |
| NC | A21 | C21 | NC |
| NC | A22 | C22 | PS* |
| NC | A23 | C23 | IDGT* |
| BD[7] | A24 | C24 | IDRQ* |
| BD[6] | A25 | C25 | BL* |
| BD[5] | A26 | C26 | 0V |
| BD[4] | A27 | C27 | NC |
| BD[3] | A28 | C28 | NC |
| BD[2] | A29 | C29 | REF8M |
| BD[1] | A30 | C30 | +5V |
| BD[0] | A31 | C31 | NC |
| +5V | A32 | C32 | NC |

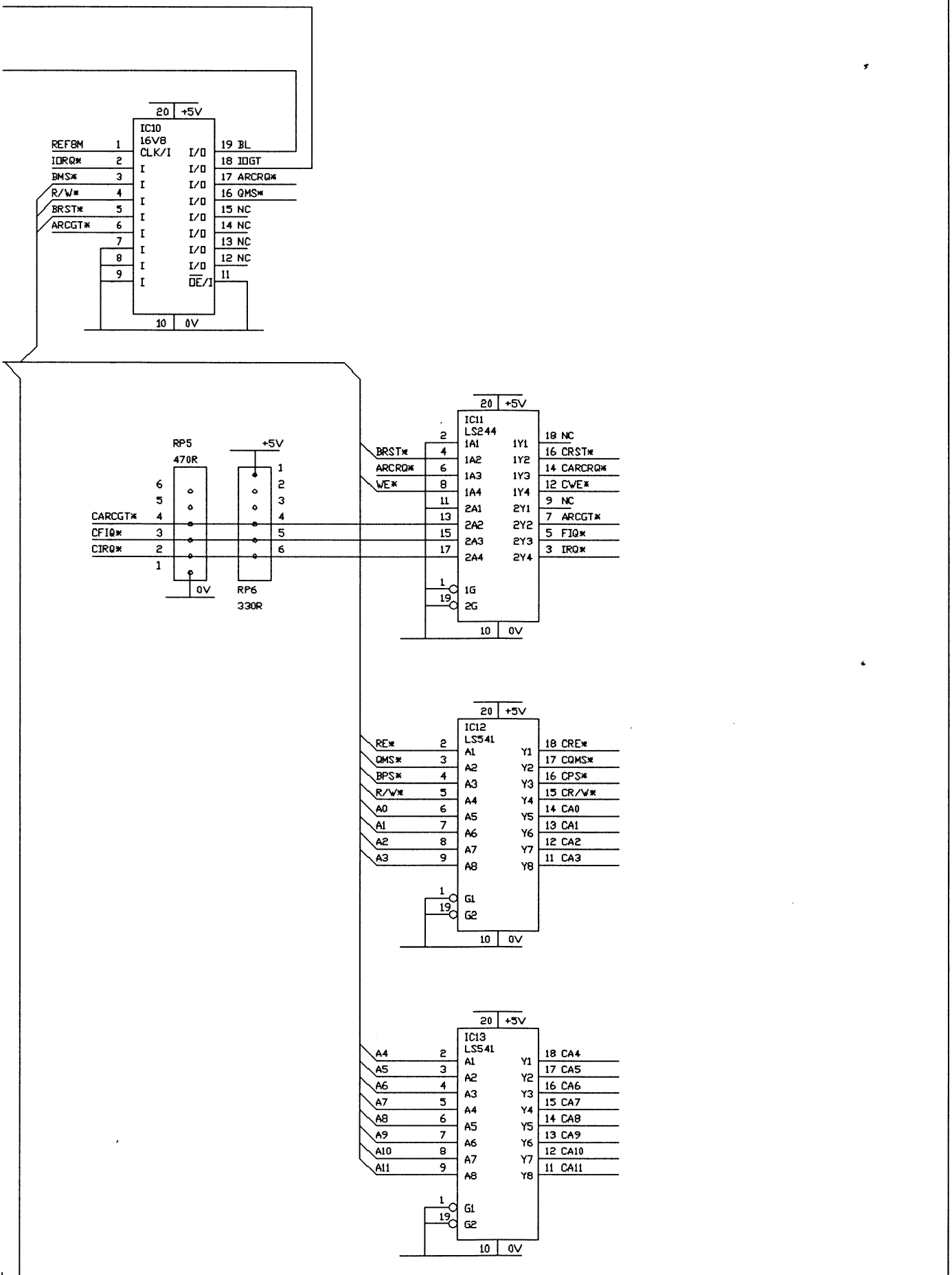
| | | |
|---|---|---|
| | | |
| | | |
| | | |
| 1 | 18 JUN 90 | A G RAY |
| ISS | DATE | DRAWN |

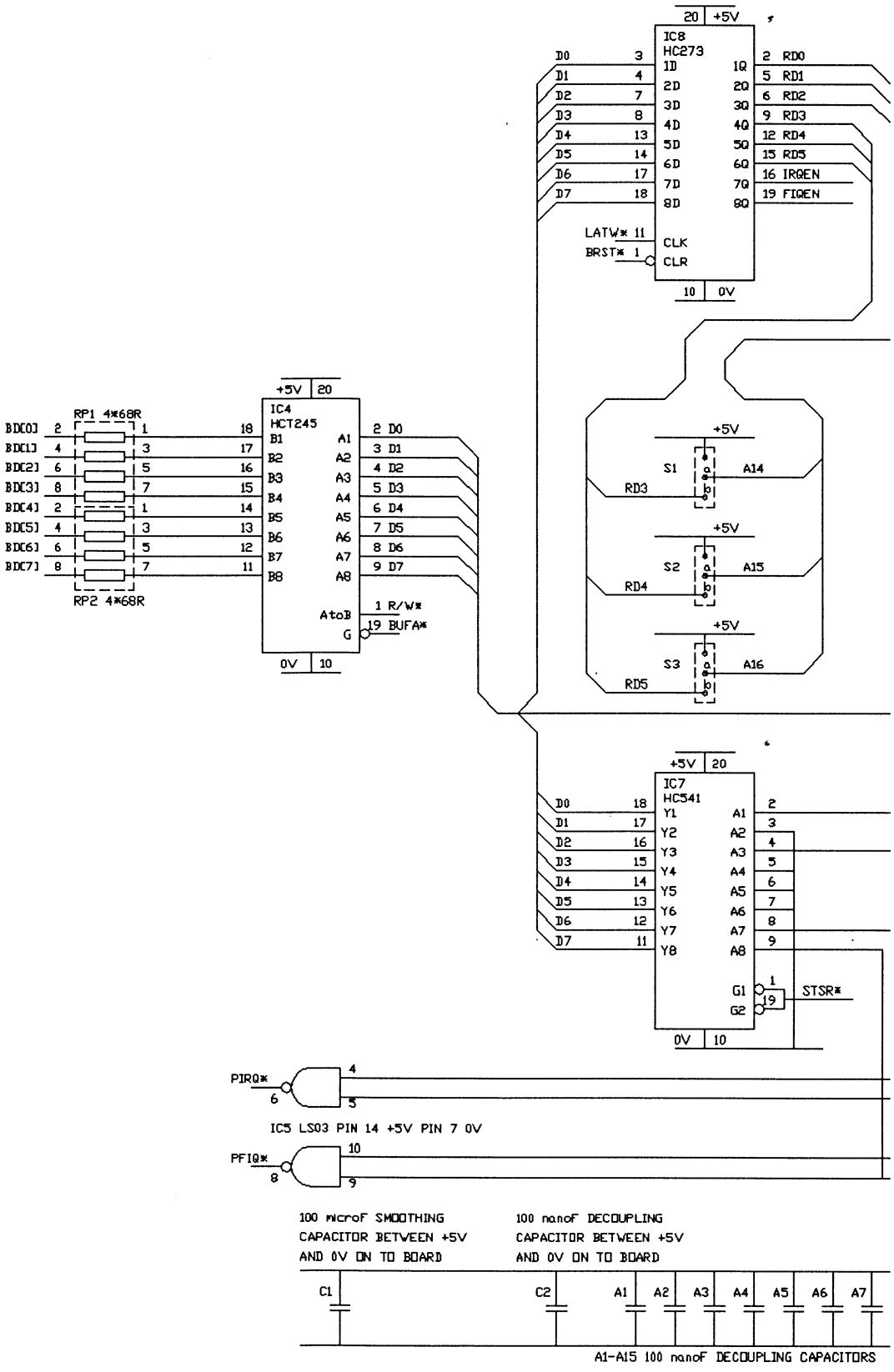


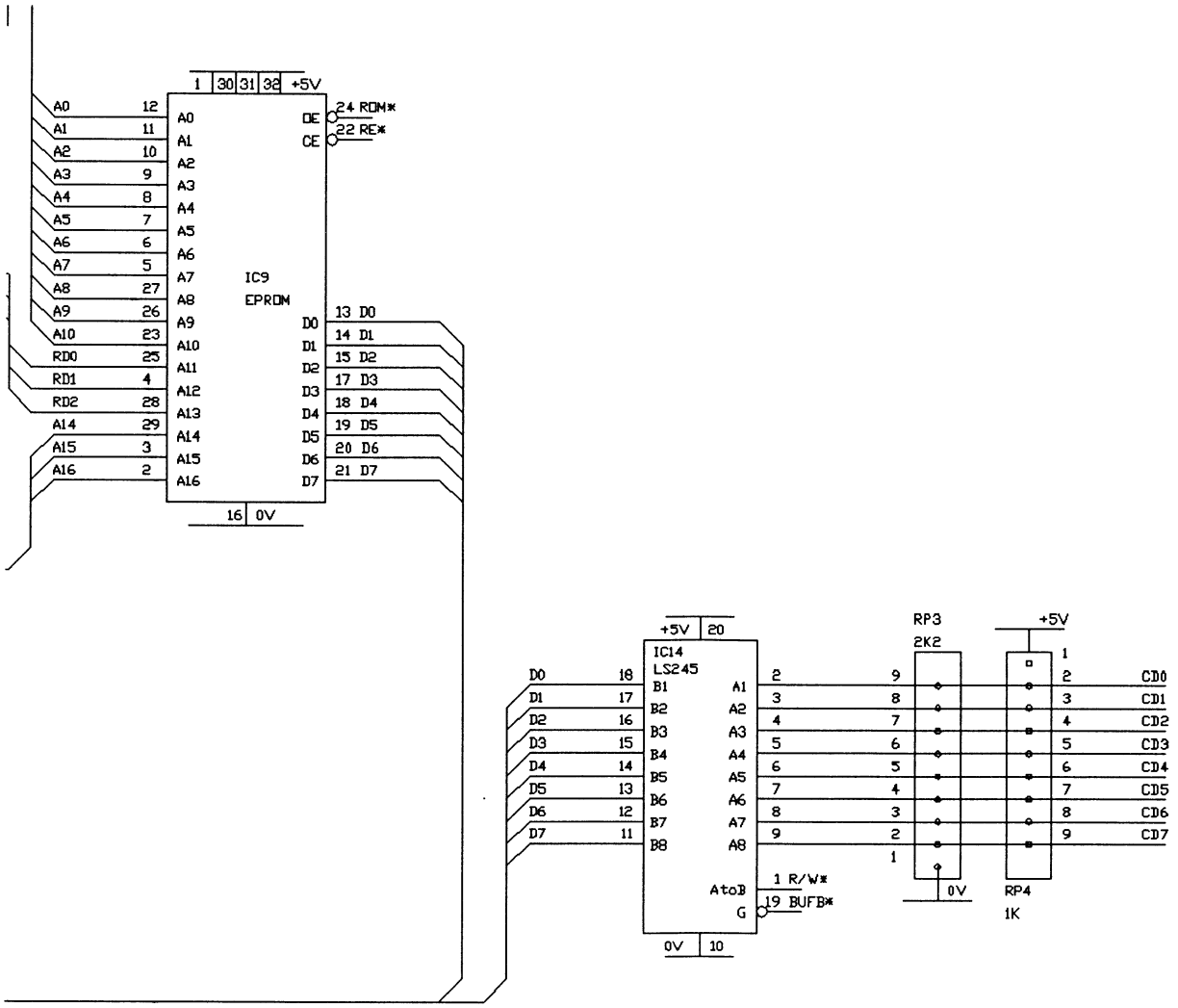
TITLE STE BUS INTERFACE
ARCHIMEDES EXPANSION CARD

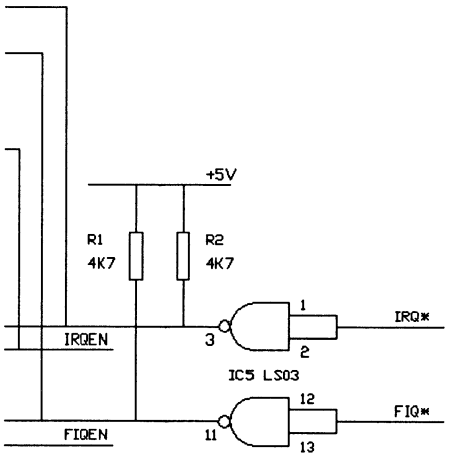
DRG NO II112.000 SHT 1 OF 3

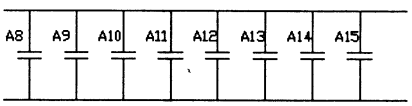

IDGT*
BL*
IC6 LS03 PIN 14 +5V PIN 7 0V
+5v
IC15
16V8
CLK/I
A10
A11
BMS*
BPS*
RE*
WE*
I/O
OE/I
19 BUFA*
18 BUFB*
17 STSR*
16 LATW*
15 ROM*
14 NC
13 NC
12 NC
11
IC1
HC541
LA[12]
LA[13]
18 A10
17 A11
16 NC
15 NC
14 NC
13 NC
12 NC
11 NC
G1
G2
IC2
HC541
LA[5]
LA[6]
LA[7]
LA[8]
LA[9]
LA[10]
MS*
LA[11]
18 A3
17 A4
16 A5
15 A6
14 A7
13 A8
12 BMS*
11 A9
IC3
HC541
PS*
PRE*
LA[2]
PWE*
LA[3]
PR/W*
LA[4]
RST*
18 BPS*
17 RE*
16 A0
15 WE*
14 A1
13 R/W*
12 A2
11 BRST*
+5V
0V


IC10
16V8
20 +5V
10 0V
REF8M 1
IDRQ* 2
BMS* 3
R/W* 4
BRST* 5
ARCGT* 6
7
8
9
CLK/I
I
I/O
OE/I
19 BL
18 IDGT
17 ARCRQ*
16 QMS*
15 NC
14 NC
13 NC
12 NC
11
RP5
470R
RP6
330R
+5V
0V
CARCGT* 4
CFIQ* 3
CIRQ* 2
1
2
3
4
5
6
IC11
LS244
BRST* 4
ARCRQ* 6
WE* 8
1A1
1A2
1A3
1A4
2A1
2A2
2A3
2A4
1G
2G
1Y1
1Y2
1Y3
1Y4
2Y1
2Y2
2Y3
2Y4
18 NC
16 CRST*
14 CARCRQ*
12 CWE*
9 NC
7 ARCGT*
5 FIQ*
3 IRQ*
IC12
LS541
RE* 2
QMS* 3
BPS* 4
R/W* 5
A0 6
A1 7
A2 8
A3 9
G1
G2
Y1
Y2
Y3
Y4
Y5
Y6
Y7
Y8
18 CRE*
17 CQMS*
16 CPS*
15 CR/W*
14 CA0
13 CA1
12 CA2
11 CA3
IC13
LS541
A4 2
A5 3
A6 4
A7 5
A8 6
A9 7
A10 8
A11 9
18 CA4
17 CA5
16 CA6
15 CA7
14 CA8
13 CA9
12 CA10
11 CA11


IC8
HC273
20 +5V
D0 3
D1 4
D2 7
D3 8
D4 13
D5 14
D6 17
D7 18
1D 2D 3D 4D 5D 6D 7D 8D
1Q 2Q 3Q 4Q 5Q 6Q 7Q 8Q
2 RD0
5 RD1
6 RD2
9 RD3
12 RD4
15 RD5
16 IRQEN
19 FIQEN
LATW* 11
BRST* 1
CLK
CLR
10 0V
RP1 4*68R
RP2 4*68R
BD[0] 2 1 18
BD[1] 4 3 17
BD[2] 6 5 16
BD[3] 8 7 15
BD[4] 2 1 14
BD[5] 4 3 13
BD[6] 6 5 12
BD[7] 8 7 11
IC4
HCT245
+5V 20
B1 B2 B3 B4 B5 B6 B7 B8
A1 A2 A3 A4 A5 A6 A7 A8
2 D0
3 D1
4 D2
5 D3
6 D4
7 D5
8 D6
9 D7
AtoB
1 R/W*
G
19 BUFA*
0V 10
+5V
S1
A14
RD3
S2
A15
RD4
S3
A16
RD5
IC7
HC541
+5V 20
D0 18
D1 17
D2 16
D3 15
D4 14
D5 13
D6 12
D7 11
Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8
A1 A2 A3 A4 A5 A6 A7 A8
2 3 4 5 6 7 8 9
G1 1
G2 19
STSR*
0V 10
PIRQ* 6 4 5
IC5 LS03 PIN 14 +5V PIN 7 0V
PFIQ* 8 10 9
100 microF SMOOTHING CAPACITOR BETWEEN +5V AND 0V ON TO BOARD
100 nanoF DECOUPLING CAPACITOR BETWEEN +5V AND 0V ON TO BOARD
C1
C2
A1 A2 A3 A4 A5 A6 A7
A1-A15 100 nanoF DECOUPLING CAPACITORS




| ISS | DATE | DRAWN |
|---|---|---|
| 4 | 12 DEC 90 | A G RAY |
| 3 | 30 NOV 90 | A G RAY |
| 2 | 20 SEP 90 | A G RAY |
| 1 | 18 JUN 90 | A G RAY |



TITLE STE BUS INTERFACE
ARCHIMEDES EXPANSION CARD

DRG NO II112.000 SHT 2 OF 3

PL2

CONNECTOR RIGHT ANGLE BOXED SOLDER PIN HEADER
WITH LONG EJECTOR/LATCH

| | | | |
|---|---|---|---|
| 0V | 1 | 2 | 0V |
| CIRQ* | 3 | 4 | 0V |
| CRST* | 5 | 6 | 0V |
| CFIQ* | 7 | 8 | 0V |
| CARCRQ* | 9 | 10 | 0V |
| CARCGT* | 11 | 12 | 0V |
| 0V | 13 | 14 | 0V |
| CWE* | 15 | 16 | 0V |
| CRE* | 17 | 18 | 0V |
| 0V | 19 | 20 | 0V |
| CQMS* | 21 | 22 | 0V |
| CPS* | 23 | 24 | 0V |
| CR/W* | 25 | 26 | 0V |
| CA0 | 27 | 28 | CA1 |
| CA2 | 29 | 30 | CA3 |
| CA4 | 31 | 32 | CA5 |
| CA6 | 33 | 34 | CA7 |
| CA8 | 35 | 36 | CA9 |
| CA10 | 37 | 38 | CA11 |
| 0V | 39 | 40 | 0V |
| CD0 | 41 | 42 | CD1 |
| CD2 | 43 | 44 | CD3 |
| CD4 | 45 | 46 | CD5 |
| CD6 | 47 | 48 | CD7 |
| 0V | 49 | 50 | 0V |

| | | |
|---|---|---|
| | | |
| | | |
| 3 | 30 NOV 90 | A G RAY |
| 1 | 18 JUN 90 | A G RAY |
| ISS | DATE | DRAWN |


INTELLIGENT INTERFACES LTD
43B WOOD STREET
STRATFORD-UPON-AVON
WARWICKSHIRE CV37 6JQ

TITLE STE BUS INTERFACE
ARCHIMEDES EXPANSION CARD

DRG NO II112.000 SHT 3 OF 3

PL1

CONNECTOR RIBBON CABLE

90 DEG 'CLICK' BOXED HEADER

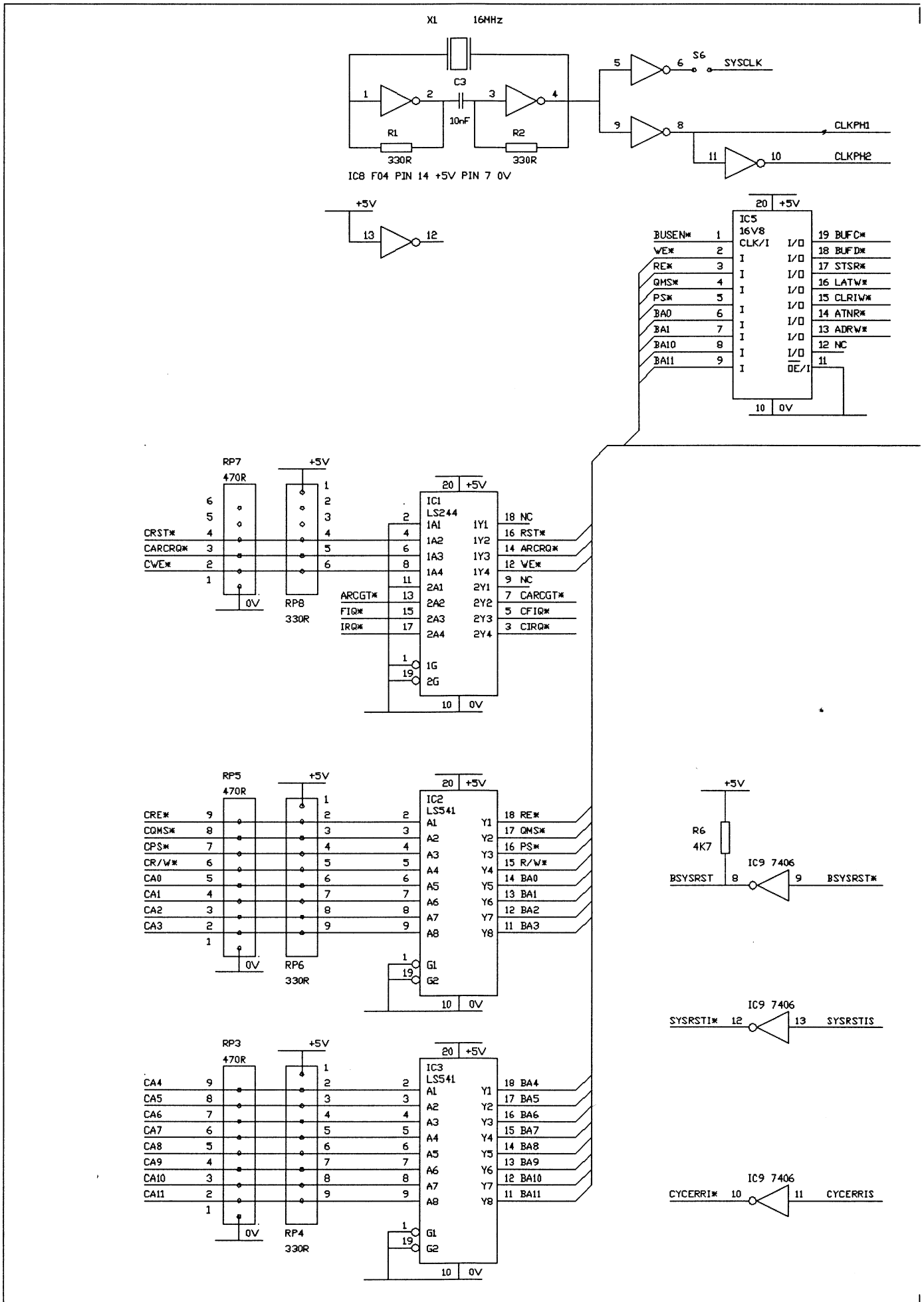
| | | | |
|---|---|---|---|
| 0V | 1 | 2 | 0V |
| CIRQ* | 3 | 4 | 0V |
| CRST* | 5 | 6 | 0V |
| CFIQ* | 7 | 8 | 0V |
| CARCRQ* | 9 | 10 | 0V |
| CARCGT* | 11 | 12 | 0V |
| 0V | 13 | 14 | 0V |
| CWE* | 15 | 16 | 0V |
| CRE* | 17 | 18 | 0V |
| 0V | 19 | 20 | 0V |
| CQMS* | 21 | 22 | 0V |
| CPS* | 23 | 24 | 0V |
| CR/W* | 25 | 26 | 0V |
| CA0 | 27 | 28 | CA1 |
| CA2 | 29 | 30 | CA3 |
| CA4 | 31 | 32 | CA5 |
| CA6 | 33 | 34 | CA7 |
| CA8 | 35 | 36 | CA9 |
| CA10 | 37 | 38 | CA11 |
| 0V | 39 | 40 | 0V |
| CD0 | 41 | 42 | CD1 |
| CD2 | 43 | 44 | CD3 |
| CD4 | 45 | 46 | CD5 |
| CD6 | 47 | 48 | CD7 |
| 0V | 49 | 50 | 0V |

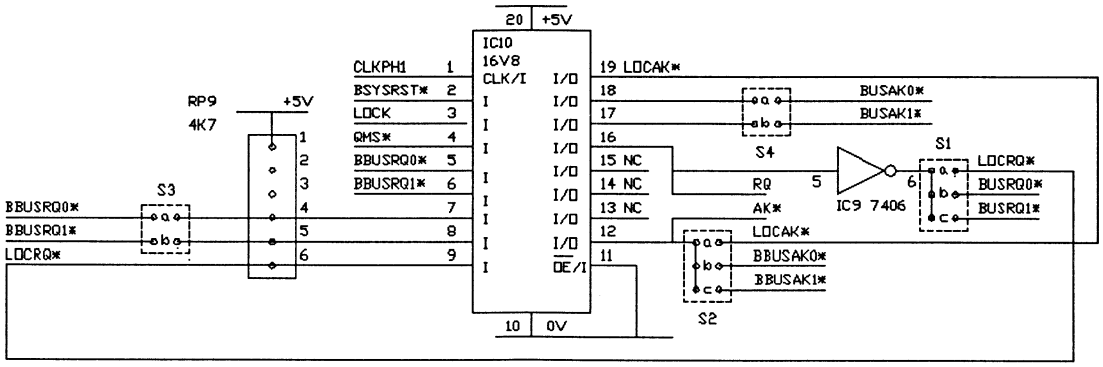
| | | |
|---|---|---|
| | | |
| | | |
| 3 | 30 NOV 90 | A G RAY |
| 1 | 22 JUN 90 | A G RAY |
| ISS | DATE | DRAWN |

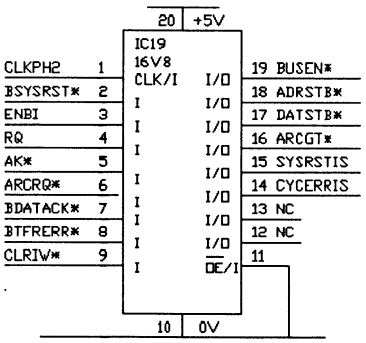

INTELLIGENT INTERFACES LTD
43B WOOD STREET
STRATFORD-UPON-AVON
WARWICKSHIRE CV37 6JQ

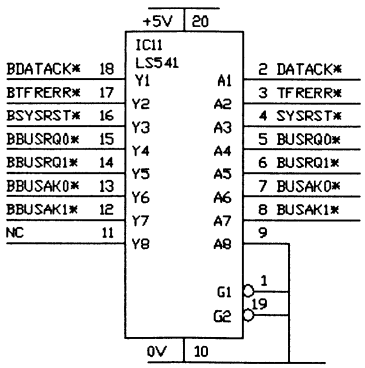
TITLE STE BUS INTERFACE
STE BUS BOARD

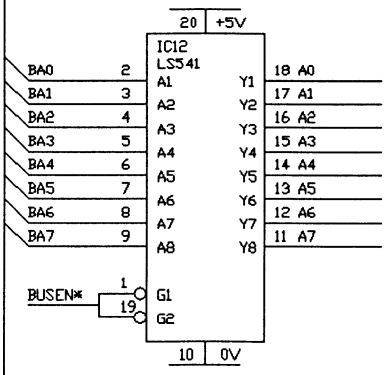
DRG NO II113.000 SHT 1 OF 3

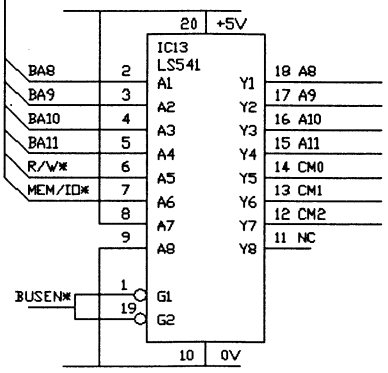

X1 16MHz
C3
10nF
R1
330R
R2
330R
S6
SYSCLK
CLKPH1
CLKPH2
IC8 F04 PIN 14 +5V PIN 7 0V
+5V
IC5
16V8
BUSEN*
WE*
RE*
QMS*
PS*
BA0
BA1
BA10
BA11
BUFC*
BUFD*
STSR*
LATW*
CLRIW*
ATNR*
ADRW*
NC
RP7
470R
RP8
330R
IC1
LS244
CRST*
CARCRQ*
CWE*
ARCGT*
FIQ*
IRQ*
RST*
ARCRQ*
WE*
CARCGT*
CFIQ*
CIRQ*
RP5
RP6
IC2
LS541
CRE*
CQMS*
CPS*
CR/W*
CA0
CA1
CA2
CA3
RE*
QMS*
PS*
R/W*
BA0
BA1
BA2
BA3
RP3
RP4
IC3
CA4
CA5
CA6
CA7
CA8
CA9
CA10
CA11
BA4
BA5
BA6
BA7
BA8
BA9
BA10
BA11
R6
4K7
IC9 7406
BSYSRST
BSYSRST*
SYSRSTI*
SYSRSTIS
CYCERRI*
CYCERRIS
0V

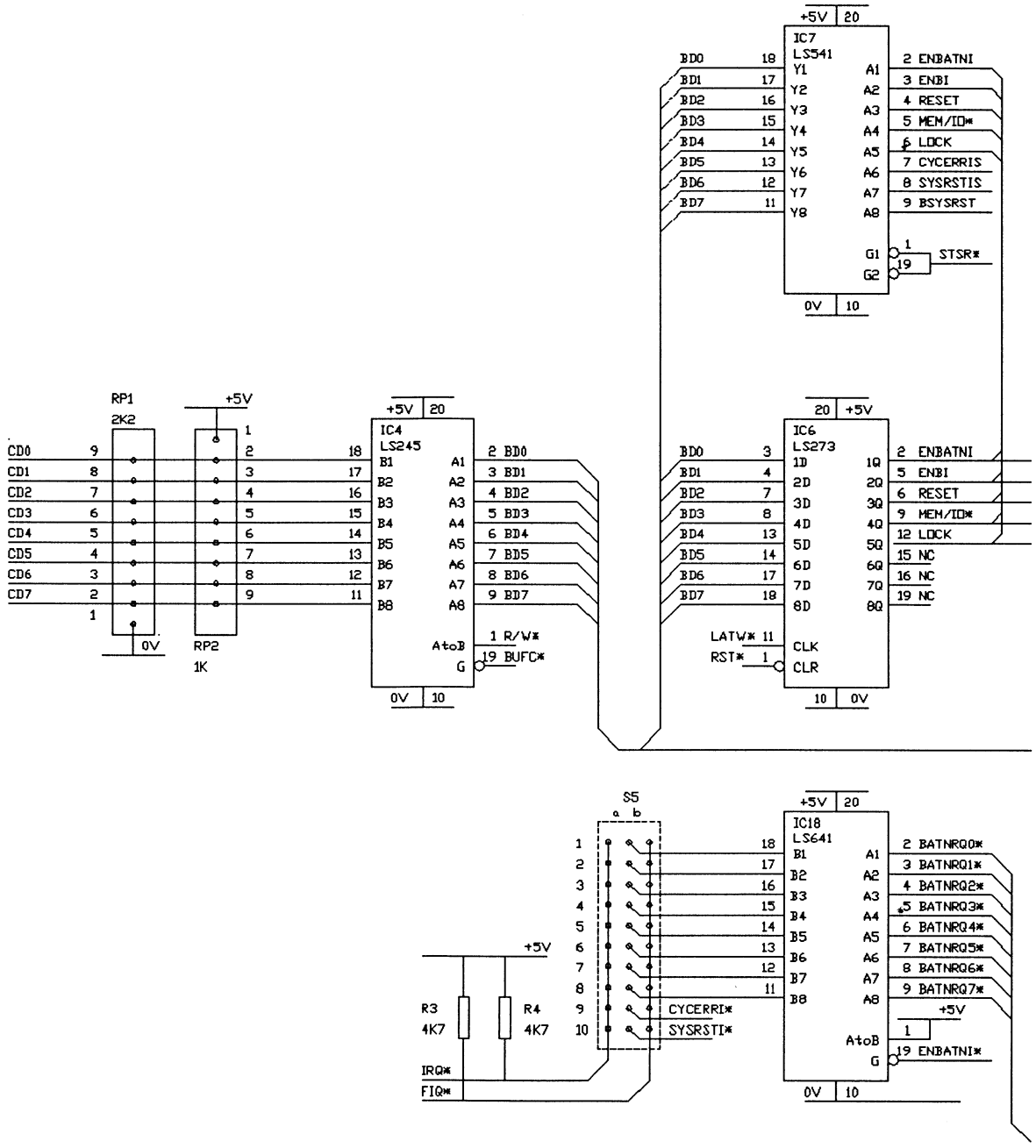

IC10
16V8
CLKPH1
BSYSRST*
LOCK
QMS*
BBUSRQ0*
BBUSRQ1*
CLK/I
I/O
OE/I
+5V
0V
19 LOCAK*
18
17
16
15 NC
14 NC
13 NC
12
11
RP9
4K7
S3
BBUSRQ0*
BBUSRQ1*
LOCRQ*
S4
BUSAK0*
BUSAK1*
RQ
AK*
IC9 7406
S1
LOCRQ*
BUSRQ0*
BUSRQ1*
S2
LOCAK*
BBUSAK0*
BBUSAK1*


IC19
16V8
CLKPH2
BSYSRST*
ENBI
RQ
AK*
ARCRQ*
BDATACK*
BTFRERR*
CLRIW*
CLK/I
I/O
OE/I
+5V
0V
19 BUSEN*
18 ADRSTB*
17 DATSTB*
16 ARCGT*
15 SYSRSTIS
14 CYCERRIS
13 NC
12 NC
11


IC11
LS541
BDATACK*
BTFRERR*
BSYSRST*
BBUSRQ0*
BBUSRQ1*
BBUSAK0*
BBUSAK1*
NC
Y1
Y2
Y3
Y4
Y5
Y6
Y7
Y8
A1
A2
A3
A4
A5
A6
A7
A8
G1
G2
+5V
0V
2 DATACK*
3 TFRERR*
4 SYSRST*
5 BUSRQ0*
6 BUSRQ1*
7 BUSAK0*
8 BUSAK1*
9

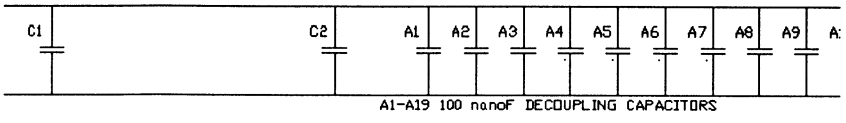

IC12
LS541
BA0
BA1
BA2
BA3
BA4
BA5
BA6
BA7
BUSEN*
A1
A2
A3
A4
A5
A6
A7
A8
Y1
Y2
Y3
Y4
Y5
Y6
Y7
Y8
G1
G2
+5V
0V
18 A0
17 A1
16 A2
15 A3
14 A4
13 A5
12 A6
11 A7

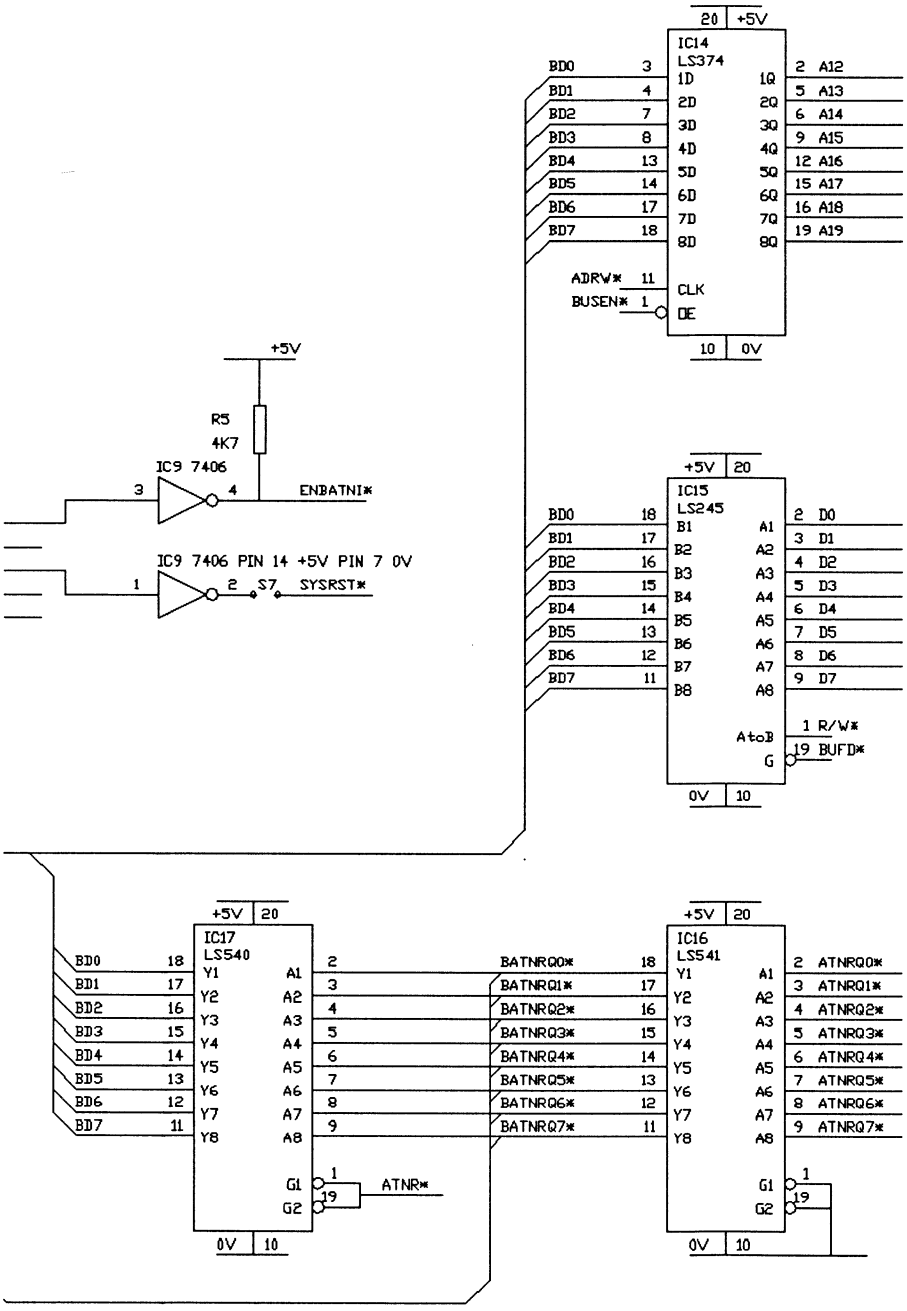

IC13
LS541
BA8
BA9
BA10
BA11
R/W*
MEM/IO*
BUSEN*
A1
A2
A3
A4
A5
A6
A7
A8
Y1
Y2
Y3
Y4
Y5
Y6
Y7
Y8
G1
G2
+5V
0V
18 A8
17 A9
16 A10
15 A11
14 CM0
13 CM1
12 CM2
11 NC

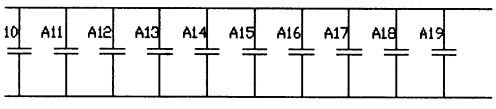



100 microF SMOOTHING CAPACITOR BETWEEN +5V AND 0V ON TO BOARD

100 nanoF DECOUPLING CAPACITOR BETWEEN +5V AND 0V ON TO BOARD






| ISS | DATE | DRAWN |
|---|---|---|
| 4 | 12 DEC 90 | A G RAY |
| 3 | 30 NOV 90 | A G RAY |
| 2 | 19 SEP 90 | A G RAY |
| 1 | 22 JUN 90 | A G RAY |



TITLE STE BUS INTERFACE
STE BUS BOARD

DRG NO II113.000 SHT 2 OF 3

PL2
CONNECTOR DIN41612 TYPE C 90 DEG PLUG
ROWS A AND C LOADED

| | | | |
|---|---|---|---|
| 0V | A1 | C1 | 0V |
| +5V | A2 | C2 | +5V |
| D0 | A3 | C3 | D1 |
| D2 | A4 | C4 | D3 |
| D4 | A5 | C5 | D5 |
| D6 | A6 | C6 | D7 |
| A0 | A7 | C7 | 0V |
| A2 | A8 | C8 | A1 |
| A4 | A9 | C9 | A3 |
| A6 | A10 | C10 | A5 |
| A8 | A11 | C11 | A7 |
| A10 | A12 | C12 | A9 |
| A12 | A13 | C13 | A11 |
| A14 | A14 | C14 | A13 |
| A16 | A15 | C15 | A15 |
| A18 | A16 | C16 | A17 |
| CM0 | A17 | C17 | A19 |
| CM2 | A18 | C18 | CM1 |
| ADRSTB* | A19 | C19 | 0V |
| DATACK* | A20 | C20 | DATSTB* |
| TFRERR* | A21 | C21 | 0V |
| ATNRQ0* | A22 | C22 | SYSRST* |
| ATNRQ2* | A23 | C23 | ATNRQ1* |
| ATNRQ4* | A24 | C24 | ATNRQ3* |
| ATNRQ6* | A25 | C25 | ATNRQ5* |
| 0V | A26 | C26 | ATNRQ7* |
| BUSRQ0* | A27 | C27 | BUSRQ1* |
| BUSAK0* | A28 | C28 | BUSAK1* |
| SYSCLK | A29 | C29 | +VSTBY |
| -12V | A30 | C30 | +12V |
| +5V | A31 | C31 | +5V |
| 0V | A32 | C32 | 0V |

| | | |
|---|---|---|
| | | |
| | | |
| | | |
| 1 | 22 JUN 90 | A G RAY |
| ISS | DATE | DRAWN |



TITLE STE BUS INTERFACE
STE BUS BOARD

DRG NO II113.000 SHT 3 OF 3