8/25/83
EJS

# BUS CONTROL PROCESS

## ACTIVITIES

1. EMPTY IR BUFFER TO { HOST RESIDENT PROCESS / SERIAL BUS / IR OUTPUT * }
2. EMPTY SERIAL IN BUFFER TO { IR / HOST RESIDENT PROCESS }
3. EMPTY ONBOARD OUT { TO SERIAL OUT / TO IR OUT }

OR

FILL INPUT TO ONBOARD from
① OUTPUT FROM ONBOARD
② IR
③ SERIAL
(flag when each has next)

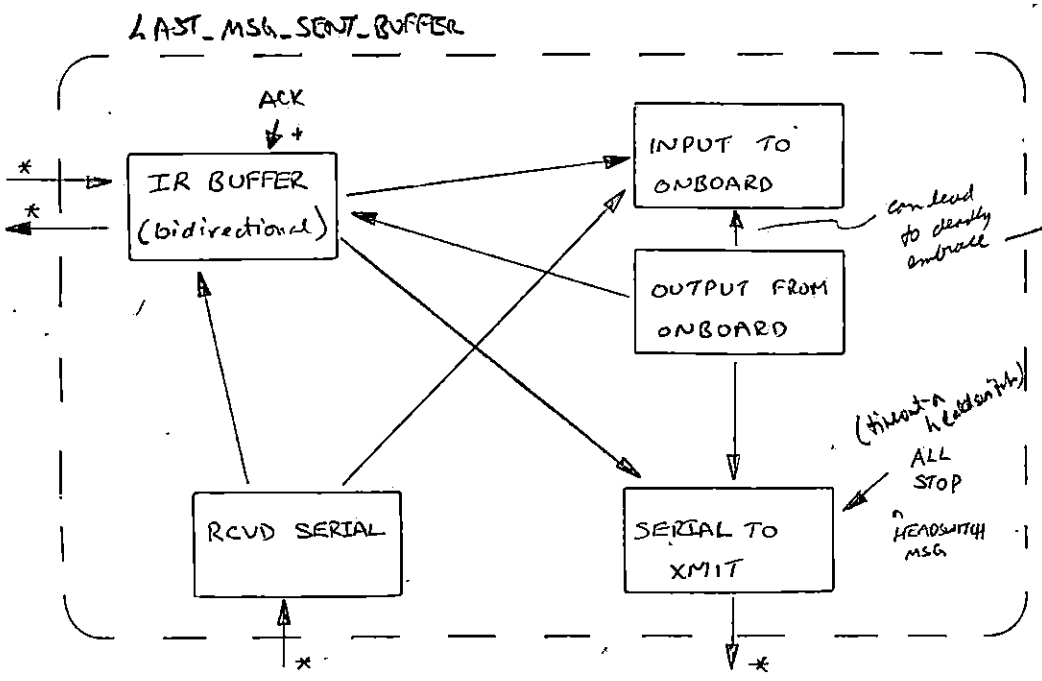
FILL SERIAL TO XMIT from
① IR
② ONBOARD

FILL IR XMIT from
① RCVD SERIAL
② OUTPUT FROM ONBOARD
③ ACK

## RESOURCES

1. SERIAL BUS OUTPUT
2. IR XMIT

((When a buffer fills, try to empty it
" " " empties, try to fill it))




could lead to deadly embrace

Unless all on board processes release input before requesting output

4 char to echo takes how long?

* DONE BY AN INTERRUPT ROUTINE

## TIMERS:

0. MODE 3
   HIGH PART IS 256 μs · MASTER INTERRUPT
   LOW PART IS USED FOR PRIORITY OK CONTROL — NO - UNUSED
1. MODE 2
   SMOD = 1, TH1 = $FD_{16}$, $C/\overline{T}$ = 0, GATE = 0

## INTERRUPTS:

$\overline{INT0}$ BUS REQUEST - ENABLED WHENEVER BUS MASTER DOES NOT NEED BUS

TIMER0 256 μs

($\overline{INT1}$) IR INPUT - (NOT) USED AS INTERRUPT - perhaps the only time critical failsafe

(TIMER1) PRIORITY OK CONTROL — NOT USED 300 ms counter

(SERIAL PORT) INTERROGATED IN 256 μs LOOP 50 ms watchdog

(H/S DEBOUNCE) DONE IN 256 μs LOOP

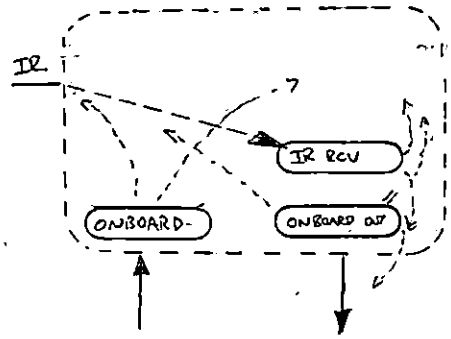
MSG BUFFERS

① If both offboard rcv buffers are full, no bus requests granted

② IR RCV gets hold of the bus before onboard out and before bus requests granted

Q. What about on board sending to speech?

③ Onboard out gets the bus before bus requests in




TOBS

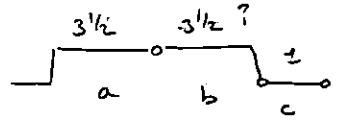
ORDER TO EMPTY:
IR RCV
ONBOARD
OFFBOARD

only one offboard buffer?
what about on board to IR?
- let it use offboard buffer?

GRANT BUS REQUEST

SET BUS IN USE BIT
(PRIORITY OK SHOULD ALREADY BE SET)
DISABLE INTØ (same?)
SET "TIMER1" 1ST INTERVAL BIT (RUNNING 1ST INTERVAL)
LOAD "TIMER1".
RUN "TIMER1"

"TIMER1"
IS "TIMER1" 1ST INTERVAL BIT SET?
YES: LOAD "TIMER 1"
RUN "TIMER1"
CLEAR 1ST INTERVAL BIT
JBC
after a
NO: IS PRIORITY OK SET?
YES: LOAD "TIMER1" w/ SHORT VALUE
CLEAR PRIORITY OK
RUN "TIMER1"
JBC
after b
NO: IS OUTBOUND MSG WAITING?
YES:
NO: ENABLE INTERRUPT
SET PRIORITY OK
after c




170 μs
50 μs

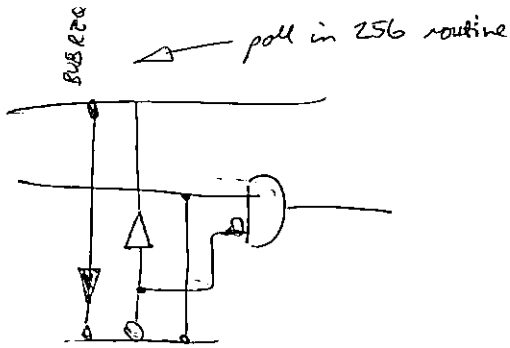
What about speech?

is buffer space available
basmath
for ~~host~~ to recv another msg?

# COMM PROCESSOR TIMING

p6-14 Timer ϕ is Mode 3 interrupt
TL is full using TF1 set

| | | |
|---|---|---|
| 256 µs | IR | TIMER ϕ (INTERRUPT ON TIMER) |
| 48 µs | SERIAL BUS | TIMER 1 - (NO INTERRUPT) |
| Variable | BUS CONTROL | ~~ISS TIMER~~ TIMER ϕ LOW |
| Random | BUS REQUEST | INTERRUPT ϕ |
| Random | SERIAL BUS ADDR | Serial INTERRUPT |
| IR | IR | INT1 (3µs) |




## TASKS

PROCESS IR MSG

PROCESS SERIAL BUS MSG

SEND OUTGOING MESSAGES

DO ONBOARD PROCESS

( H/S: status to host, on hit send msg to motors or ... )

( version:

( topo id:

(

SPEECH control

256 µs ISR

START

exclusive { SEND IR BIT IF REQD
RCV IR BIT IF REQD } 6-7 µs

exclusive { SEND IR IF REQD
RCV IR IF REQD } 1-30 µs

exclusive { SEND SERIAL BYTE IF REQD
RCV SERIAL BYTE IF REQD } some?

DEBOUNCE HEADSWITCH - 32 ms * 50 sec watchdog

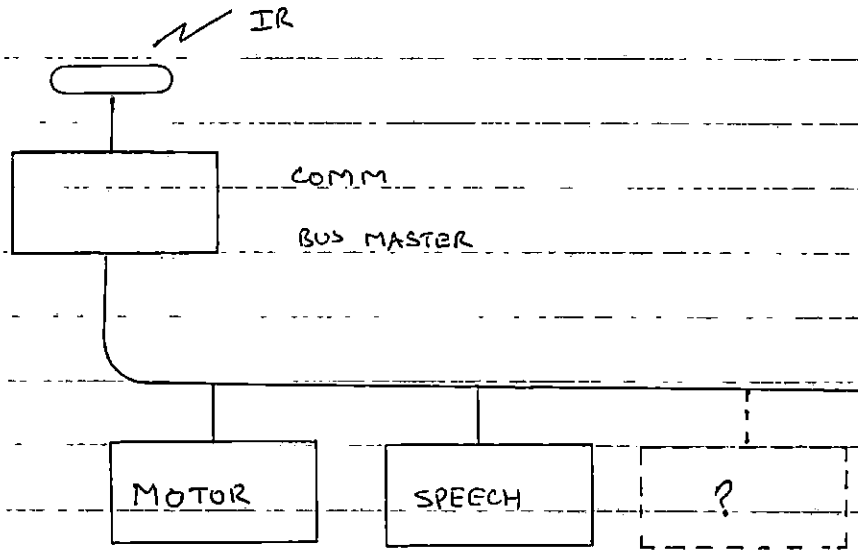
CHECK FOR SIGNAL RCVD - 320 ms

# ON BOARD TASKS

① EMPTY IR RCV BUFFER
- gets priority for serial bus over ONBOARD out & external requests
- gets software priority over all other host processes

② EMPTY ONBOARD OUT
- gets priority for IR XMIT over OFFBOARD
- loses priority for serial bus to IR XMIT
- has software priority over OFFBOARD and all host processes

③ EMPTY OFFBOARD
- has software priority over host processes




1. Initial System

bus master handles IR communications

auxiliary devices communicate only w/ host computer, and that via bus master

LIMITATIONS:

high level on board functions not supported

- collision imminent to stop motors
- wall hug to steer base
- environmentally controlled speech (battery low) (lost) (train)
- BLB CPU commit num system (?)
- stop on head switch (?)

REQMT ~~SOLN~~: auxiliary devices must be able to ~~handle~~ request bus service

POSSIBLE SOLN's:

poll devices individually via serial bus - (speed?) (how many devices? RAM?)

interrupt request - (2 wires for daisy chain)

poll devices by bus posn - (they don't know bus posn)

non-prioritized interrupt -

EW 8/19

# ON-BOARD-BUS DESIGN REQMTS

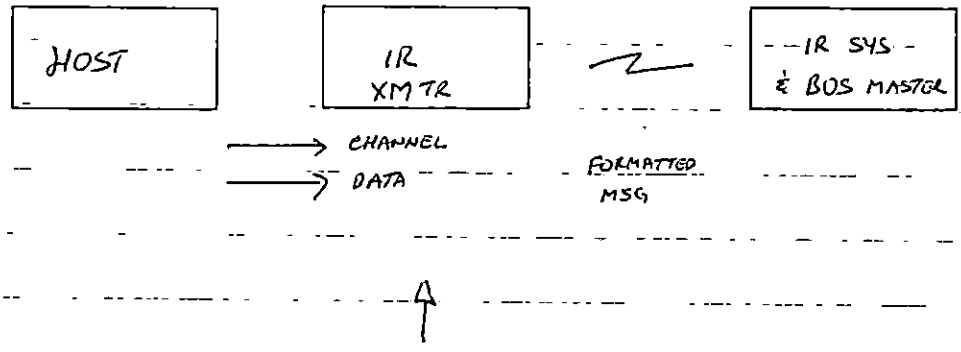
1. Support msg passing to/from host to individual bus devices
2. " " " from " " all bus devices
3. " all device reset in addition to (2)
4. " msg passing from bus device to bus device

## OBB ASSUMPTIONS

1. OBB reliability does not require error detection or correction
   - no ACKs
   - no parity
   - no checksums
2. Bus master handles all IR formatting
   - acks & sayuhots
   - parity & checksum generation/stripping

# COMPATIBILITY w/ LOWER COST IR

PRESENT




REDO OPTIONS:

1. no processing or memory (still need a bus master on board to buffer incoming since we can't run an interrupt line to host)
2. same s/w, new analog section

BANK $\phi$ serial bus recv (?)

Q. Process buffer undo? or new and overwrite?

## Serial Rcv Routine

```
        PUSH    ACC             ; SAVE ACCUM FOR INTERRUPTED PROCESS        2
        PUSH    PSW             ; AND ALSO PSW, SINCE CJNE ALTERS CARRY     2
        MOV     SBUF, ACC       ; LOOK AT RECEIVED BYTE                     2

        JZ      GET ALL CALL DATA   ; IF ITS ZERO, HANDLE AN ALL CALL       2

        CJNE    A, #PROCESSID $\phi$, CK PROCESS 1 ID ; IF ITS NOT PROCESS $\phi$ ON BOARD, CHECK PROCESS 1   3
        LJMP    CPTR PROCESS $\phi$ DATA                                    3

CK PROCESS 1 ID
        POP.    PSW             ; RESTORE PSW FOR INTERRUPTED PROCESS       2
IGNREMSG POP    ACC             ; RESTORE ACCUMULATOR FOR INTERRUPTED PROCESS 2

        RETI                    ; RETURN TO INTERRUPTED PROCESS             1

                                                                  19 µs if no
```

For most process
once a process is doing a task, it will respond to all-calls and universal msgs only. It will not respond to other msgs unless

① System reset

② process disable with a universal $\phi$2XX

③ process terminates

QESTIONS:

| | Rec |
|---|---|
| only 8 devices, as per present? (ref Topo Comm Protocol 3, p5) | NO 32-255 valid |
| 7 byte IR packet, as per present --- a same length? | YES |
| error correction on board w/ parity & checksum? | YES |
| on a message to device X, how does the bus master know if a response is requested... and how quickly? | DOESN'T |
| compatibility w/ deeper IR system? | |
| rotating vs fixed order priority? needs extra line for IR to get the bus whenever | |
| does the master care who is requesting the bus? | |

PROPOSALS

1. leave the HOST/DL/IR protocol alone; (ack, say what, etc)
2. message contents are ~~or the protocol~~ a will change, incl. device numbers
3. each data byte has ~~valid~~ legal range of 32-255 on the ~~host~~ host/DL/IR portion
4.

DEVICE NUMBER ASS.

| | |
|---|---|
| 0 | ~~host computer via~~ host via ~~bus master to right~~ IR system (send it out) |
| 1 | bus master |
| 2 | all devices ~~w/ default~~ (reset is hardwired) |
| 3 | ~~give system~~ id after system via IR (fridge, etc). |
| | |
| 31 | |
| 32 | motors |
| 33 | speech |
| | ⋮ |
| 255 | |