

Programming Project Checkpoint 4

111062311 林哲宇

1. Screenshot for compilation:

```
jeffreylin0909@DESKTOP-Q29MBHF:/mnt/c/Users/林哲宇/OneDrive/桌面/OS/OS check point 4/ppc4$ make
sdcc -c test3threads.c
test3threads.c:106: warning 158: overflow in implicit constant conversion
sdcc -c preemptive.c
preemptive.c:206: warning 85: in function ThreadCreate unreferenced function argument : 'fp'
sdcc -o test3threads.hex test3threads.rel preemptive.rel
jeffreylin0909@DESKTOP-Q29MBHF:/mnt/c/Users/林哲宇/OneDrive/桌面/OS/OS check point 4/ppc4$ ls
Makefile      preemptive.h    preemptive.rst  test3threads.c  test3threads.lst test3threads.rel
preemptive.asm preemptive.lst  preemptive.sym  test3threads.hex test3threads.map test3threads.rst
preemptive.c   preemptive.rel  test3threads.asm test3threads.lk  test3threads.mem test3threads.sym
jeffreylin0909@DESKTOP-Q29MBHF:/mnt/c/Users/林哲宇/OneDrive/桌面/OS/OS check point 4/ppc4$ |
```

→ The warning message is because of setting TH1 to negative number (-6 for UART baud rate), and it's totally safe.

→ The warning message is because of using DPH/DPL instead of identifier of parameter, and it's totally safe.

→ Files generated after compilation (including .hex and .map).

Note: for better understanding for following explanation, here's the variable address map of my code:

00000020	_T_SP (20~23, 1 address space each entry)	preemptive
00000024	_current_T	preemptive
00000025	_tmp0	preemptive
00000026	_tmp1	preemptive
00000027	_tmp2	preemptive
00000028	_bitmap	preemptive
00000029	_mutex	test3threads
0000002A	_full	test3threads
0000002B	_empty	test3threads
0000002C	_b_start	test3threads
0000002D	_b_end	test3threads
0000002E	_in_counter_1	test3threads
0000002F	_in_counter_2	test3threads
00000030	_buffer(30~32, 1 address space each entry)	test3threads
00000033	__1_turn	test3threads
00000034	__2_turn	test3threads

(semaphores to do classical bounded buffer)

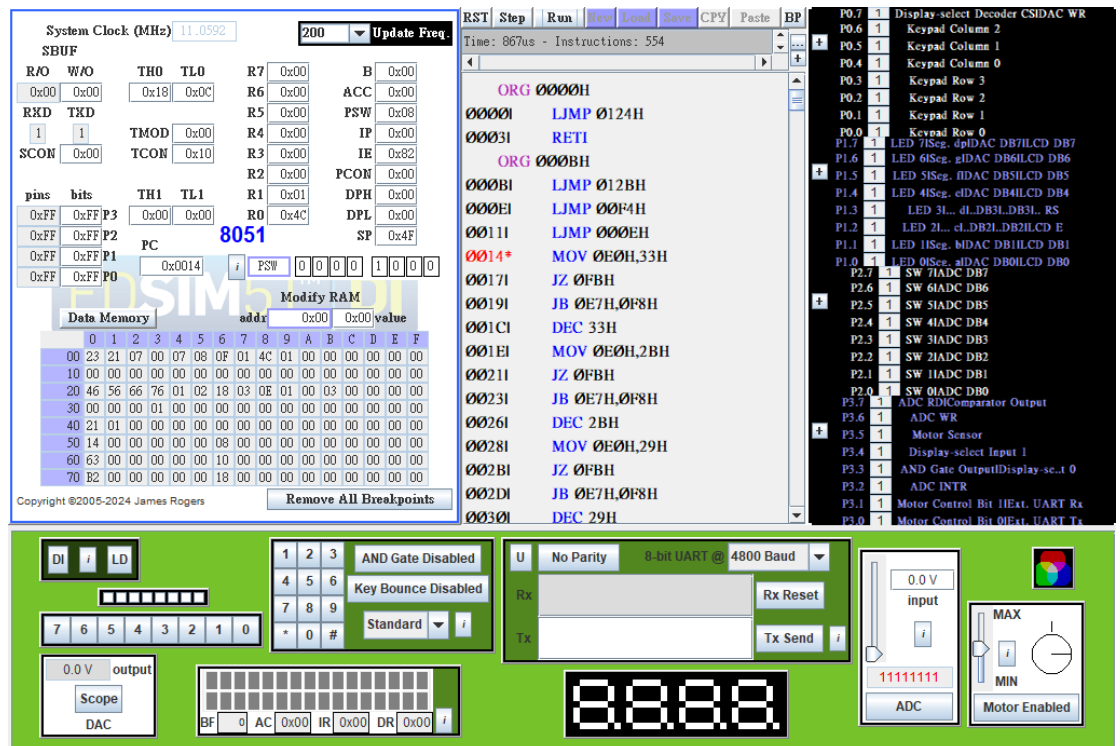
(semaphores to make fair)

And here's the function address map:

00000014	_Producer1	test3threads
00000063	_Producer2	test3threads
000000B2	_Consumer	test3threads
000000F4	_main	test3threads
00000124	__sdcc_gsinit_startup	test3threads
00000128	__mcs51_genRAMCLEAR	test3threads
00000129	__mcs51_genXINIT	test3threads
0000012A	__mcs51_genXRAMCLEAR	test3threads
0000012B	_timer0_ISR	test3threads
0000012F	_Bootstrap	preemptive
00000161	_ThreadCreate	preemptive
000001F3	_ThreadYield	preemptive
00000248	_ThreadExit	preemptive
000002A1	_myTimer0Handler	preemptive

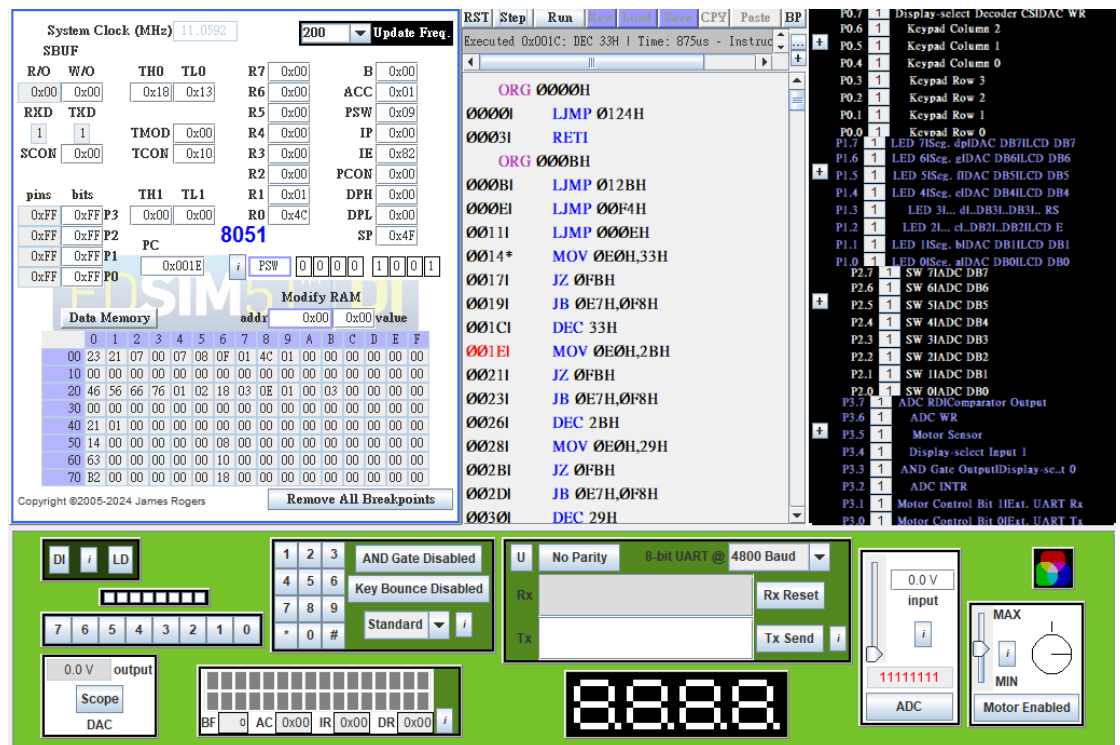
2. Screenshots and explanation:

- Producer1:



▲ Just enter Producer1

(mutex=1, full=0, empty=3, _1_turn=1, _2_turn=0).



▲ After SemaphoreWait(_1_turn)

(mutex=1, full=0, empty=3, _1_turn=0, _2_turn=0).

The screenshot shows the Proteus ISIS simulation environment. The assembly code editor is open, displaying the following code:

```

ORG 0000H
00000 LJMP 0124H
00003 RETI
ORG 000BH
000B1 LJMP 012BH
000E1 LJMP 00F4H
00111 LJMP 000EH
0014* MOV 0E0H,33H
00171 JZ 0FBH
00191 JB 0E7H,0F8H
001C1 DEC 33H
001E1 MOV 0E0H,2BH
00211 JZ 0FBH
00231 JB 0E7H,0F8H
00261 DEC 2BH
00281 MOV 0E0H,29H
002B1 JZ 0FBH
002D1 JB 0E7H,0F8H
00301 DEC 29H
  
```

The I/O components are configured as follows:

- DI: 7, LD
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 0.0 V output
- Scope
- DAC
- BF: 0, AC: 0x00, IR: 0x00, DR: 0x00
- 8-bit UART @ 4800 Baud
- Rx Reset
- Tx Send
- 0.0 V input
- 11111111
- ADC
- MAX
- MIN
- Motor Enabled

The logic analyzer shows the following data:

Address	Value
00	23
01	07
02	00
03	07
04	08
05	0F
06	01
07	4C
08	01
09	00
0A	00
0B	00
0C	00
0D	00
0E	00
0F	00
10	00
11	00
12	00
13	00
14	00
15	00
16	00
17	00
18	00
19	00
1A	00
1B	00
1C	00
1D	00
1E	00
1F	00
20	46
21	56
22	66
23	76
24	01
25	02
26	18
27	03
28	0E
29	01
2A	00
2B	02
2C	00
2D	00
2E	00
2F	00
30	00
31	00
32	00
33	00
34	00
35	00
36	00
37	00
38	00
39	00
3A	00
3B	00
3C	00
3D	00
3E	00
3F	00
40	21
41	01
42	00
43	00
44	00
45	00
46	00
47	00
48	00
49	00
4A	00
4B	00
4C	00
4D	00
4E	00
4F	00
50	14
51	00
52	00
53	00
54	00
55	00
56	00
57	00
58	00
59	00
5A	00
5B	00
5C	00
5D	00
5E	00
5F	00
60	63
61	00
62	00
63	00
64	00
65	00
66	00
67	00
68	00
69	00
6A	00
6B	00
6C	00
6D	00
6E	00
6F	00
70	82
71	00
72	00
73	00
74	00
75	00
76	00
77	00
78	00
79	00
7A	00
7B	00
7C	00
7D	00
7E	00
7F	00

▲ After SemaphoreWait (empty)
(mutex=1, full=0, empty=2, _1_turn=0, _2_turn=0).

The screenshot shows the Proteus ISIS simulation environment. The assembly code editor is open, displaying the following code:

```

000E1 LJMP 00F4H
00111 LJMP 000EH
0014* MOV 0E0H,33H
00171 JZ 0FBH
00191 JB 0E7H,0F8H
001C1 DEC 33H
001E1 MOV 0E0H,2BH
00211 JZ 0FBH
00231 JB 0E7H,0F8H
00261 DEC 2BH
00281 MOV 0E0H,29H
002B1 JZ 0FBH
002D1 JB 0E7H,0F8H
00301 DEC 29H
00321 CLR 0AFH
00341 MOV A,2DH
00361 ADD A,#30H
00381 MOV R1,A
00391 MOV R7,2EH
  
```

The I/O components are configured as follows:

- DI: 7, LD
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 0.0 V output
- Scope
- DAC
- BF: 0, AC: 0x00, IR: 0x00, DR: 0x00
- 8-bit UART @ 4800 Baud
- Rx Reset
- Tx Send
- 0.0 V input
- 11111111
- ADC
- MAX
- MIN
- Motor Enabled

The logic analyzer shows the following data:

Address	Value
00	23
01	07
02	00
03	07
04	08
05	0F
06	01
07	4C
08	01
09	00
0A	00
0B	00
0C	00
0D	00
0E	00
0F	00
10	00
11	00
12	00
13	00
14	00
15	00
16	00
17	00
18	00
19	00
1A	00
1B	00
1C	00
1D	00
1E	00
1F	00
20	46
21	56
22	66
23	76
24	01
25	02
26	18
27	03
28	0E
29	01
2A	00
2B	02
2C	00
2D	00
2E	00
2F	00
30	00
31	00
32	00
33	00
34	00
35	00
36	00
37	00
38	00
39	00
3A	00
3B	00
3C	00
3D	00
3E	00
3F	00
40	21
41	01
42	00
43	00
44	00
45	00
46	00
47	00
48	00
49	00
4A	00
4B	00
4C	00
4D	00
4E	00
4F	00
50	14
51	00
52	00
53	00
54	00
55	00
56	00
57	00
58	00
59	00
5A	00
5B	00
5C	00
5D	00
5E	00
5F	00
60	63
61	00
62	00
63	00
64	00
65	00
66	00
67	00
68	00
69	00
6A	00
6B	00
6C	00
6D	00
6E	00
6F	00
70	82
71	00
72	00
73	00
74	00
75	00
76	00
77	00
78	00
79	00
7A	00
7B	00
7C	00
7D	00
7E	00
7F	00

▲ After SemaphoreWait (mutex)
(mutex=0, full=0, empty=2, _1_turn=0, _2_turn=0).

System Clock (MHz): 11.0592

200 Update Freq.

Executed 0x005B: INC 29H | Time: 915us - Instruc

00461 CJNE A,2DH,03H

00491 MOV 2DH,#00H

004C1 MOV A,2EH

004E1 INC A

004F1 MOV 2EH,A

00511 MOV A,#1AH

00531 CJNE A,2EH,03H

00561 MOV 2EH,#00H

00591 SETB 0AFH

005B1 INC 29H

005D1 INC 2AH

005F1 INC 34H

00611 SJMP 0B1H

00631 MOV 0E0H,34H

00661 JZ 0FBH

00681 JB 0E7H,0F8H

006B1 DEC 34H

006D1 MOV 0E0H,2BH

00701 JZ 0FBH

DI LD

AND Gate Disabled

Key Bounce Disabled

Standard

0.0 V output

Scope DAC

0.0 V input

11111111

ADC

Motor Enabled

▲ After SemaphoreSignal (mutex)

(mutex=1, full=0, empty=2, _1_turn=0, _2_turn=0).

System Clock (MHz): 11.0592

200 Update Freq.

Executed 0x005D: INC 2AH | Time: 916us - Instruc

00461 CJNE A,2DH,03H

00491 MOV 2DH,#00H

004C1 MOV A,2EH

004E1 INC A

004F1 MOV 2EH,A

00511 MOV A,#1AH

00531 CJNE A,2EH,03H

00561 MOV 2EH,#00H

00591 SETB 0AFH

005B1 INC 29H

005D1 INC 2AH

005F1 INC 34H

00611 SJMP 0B1H

00631 MOV 0E0H,34H

00661 JZ 0FBH

00681 JB 0E7H,0F8H

006B1 DEC 34H

006D1 MOV 0E0H,2BH

00701 JZ 0FBH

DI LD

AND Gate Disabled

Key Bounce Disabled

Standard

0.0 V output

Scope DAC

0.0 V input

11111111

ADC

Motor Enabled

▲ After SemaphoreSignal (full)

(mutex=1, full=1, empty=2, _1_turn=0, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	0x00	B	0x00
0x00	0x00	0x19	0x1A	R6	0x00	ACC	0x1A
RKD	TKD			R5	0x00	PSW	0x09
1	1	TMOD	0x00	R4	0x00	IP	0x00
SCON	0x00	TCON	0x10	R3	0x00	IE	0x82
				R2	0x00	PCON	0x00
pins	bits	TH1	TL1	R1	0x30	DPH	0x00
0xFF	0xFF	P3	0x00	R0	0x4C	DPL	0x00
0xFF	0xFF	P2				SP	0x4F
0xFF	0xFF	P1					
0xFF	0xFF	P0					

PC: 8051

Modify RAM

Data Memory	addr	0x00	0x00	value												
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00	23	21	07	00	07	08	0F	01	4C	30	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	46	56	66	76	01	02	18	03	0E	01	01	02	00	01	01	00
30	41	00	00	00	01	00	00	00	00	00	00	00	00	00	00	00
40	21	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	14	00	00	00	00	00	08	00	00	00	00	00	00	00	00	00
60	63	00	00	00	00	00	10	00	00	00	00	00	00	00	00	00
70	B2	00	00	00	00	00	18	00	00	00	00	00	00	00	00	00

Copyright ©2005-2024 James Rogers

Remove All Breakpoints

Assembly Code:

```

000461 CJNE A,2DH,03H
000491 MOV 2DH,#00H
0004C1 MOV A,2EH
0004E1 INC A
0004F1 MOV 2EH,A
000511 MOV A,#1AH
000531 CJNE A,2EH,03H
000561 MOV 2EH,#00H
000591 SETB 0AFH
0005B1 INC 29H
0005D1 INC 2AH
0005F1 INC 34H
000611 SJMP 0B1H
000631 MOV 0E0H,34H
000661 JZ 0FBH
000681 JB 0E7H,0F8H
0006B1 DEC 34H
0006D1 MOV 0E0H,2BH
000701 JZ 0FBH
  
```

Hardware Interface:

- DI: 1, LD: 1
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 8-bit UART @ 4800 Baud
- No Parity
- Rx Reset
- Tx Send
- 0.0 V output
- Scope
- DAC
- ADC: 11111111
- Motor Enabled

▲ After SemaphoreSignal(_2_turn)
(mutex=1, full=1, empty=2, _1_turn=0, _2_turn=1).

● Producer2:

System Clock (MHz): 11.0592 | 200 | Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	0x00	B	0x00
0x00	0x00	0x02	0x02	R6	0x00	ACC	0x00
RKD	TKD			R5	0x00	PSW	0x10
1	1	TMOD	0x00	R4	0x00	IP	0x00
SCON	0x00	TCON	0x10	R3	0x00	IE	0x82
				R2	0x00	PCON	0x00
pins	bits	TH1	TL1	R1	0x00	DPH	0x00
0xFF	0xFF	P3	0x00	R0	0x00	DPL	0x00
0xFF	0xFF	P2				SP	0x5F
0xFF	0xFF	P1					
0xFF	0xFF	P0					

PC: 8051

Modify RAM

Data Memory	addr	0x00	0x00	value												
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00	23	21	07	00	07	08	0F	01	21	22	00	00	00	00	02	02
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	46	56	66	76	02	04	18	03	0E	01	01	02	00	01	01	00
30	41	00	00	00	01	00	00	00	00	00	00	00	00	00	00	00
40	21	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	14	00	00	00	00	00	08	00	00	00	00	00	00	00	00	00
60	63	00	00	00	00	00	10	00	00	00	00	00	00	00	00	00
70	B2	00	00	00	00	00	18	00	00	00	00	00	00	00	00	00

Copyright ©2005-2024 James Rogers

Remove All Breakpoints

Assembly Code:

```

000611 SJMP 0B1H
000631 MOV 0E0H,34H
000661 JZ 0FBH
000681 JB 0E7H,0F8H
0006B1 DEC 34H
0006D1 MOV 0E0H,2BH
000701 JZ 0FBH
000721 JB 0E7H,0F8H
000751 DEC 2BH
000771 MOV 0E0H,29H
0007A1 JZ 0FBH
0007C1 JB 0E7H,0F8H
0007F1 DEC 29H
000811 CLR 0AFH
000831 MOV A,2DH
000851 ADD A,#30H
000871 MOV R1,A
000881 MOV R7,2FH
0008A1 MOV A,#30H
  
```

Hardware Interface:

- DI: 1, LD: 1
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 8-bit UART @ 4800 Baud
- No Parity
- Rx Reset
- Tx Send
- 0.0 V output
- Scope
- DAC
- ADC: 11111111
- Motor Enabled

▲ Just enter Producer2

(mutex=1, full=1, empty=2, _1_turn=0, _2_turn=1).

The screenshot displays the Proteus simulation environment. The assembly window shows the following code:

```

00611 SJMP 0B1H
0063* MOV 0E0H,34H
00661 JZ 0FBH
00681 JB 0E7H,0F8H
006B1 DEC 34H
006D1 MOV 0E0H,2BH
00701 JZ 0FBH
00721 JB 0E7H,0F8H
00751 DEC 2BH
00771 MOV 0E0H,29H
007A1 JZ 0FBH
007C1 JB 0E7H,0F8H
007F1 DEC 29H
00811 CLR 0AFH
00831 MOV A,2DH
00851 ADD A,#30H
00871 MOV R1,A
00881 MOV R7,2FH
008A1 MOV A,#30H
  
```

The hardware interface at the bottom shows a keypad, a 7-segment display showing '8888', an ADC input set to 0.0V, and a motor control indicator labeled 'Motor Enabled'.

▲ After SemaphoreWait(_2_turn)

(mutex=1, full=1, empty=2, _1_turn=0, _2_turn=0).

The screenshot displays the Proteus simulation environment. The assembly window shows the following code:

```

00591 SETB 0AFH
005B1 INC 29H
005D1 INC 2AH
005F1 INC 34H
00611 SJMP 0B1H
0063* MOV 0E0H,34H
00661 JZ 0FBH
00681 JB 0E7H,0F8H
006B1 DEC 34H
006D1 MOV 0E0H,2BH
00701 JZ 0FBH
00721 JB 0E7H,0F8H
00751 DEC 2BH
00771 MOV 0E0H,29H
007A1 JZ 0FBH
007C1 JB 0E7H,0F8H
007F1 DEC 29H
00811 CLR 0AFH
00831 MOV A,2DH
  
```

The hardware interface at the bottom shows a keypad, a 7-segment display showing '8888', an ADC input set to 0.0V, and a motor control indicator labeled 'Motor Enabled'.

▲ After SemaphoreWait(empty)

(mutex=1, full=1, empty=1, _1_turn=0, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

Executed 0x00AC: INC 2AH | Time: 9ms 30us - Inst

Assembly Code:

```

00931 MOV A, #03H
00951 CJNE A, 2DH, #03H
00981 MOV 2DH, #00H
009B1 MOV A, 2FH
009D1 INC A
009E1 MOV 2FH, A
00A01 MOV A, #0AH
00A21 CJNE A, 2FH, #03H
00A51 MOV 2FH, #00H
00A81 SETB 0AFH
00AA1 INC 29H
00AC1 INC 2AH
00AE1 INC 33H
00B01 SJMP 0B1H
00B21 ORL 89H, #20H
00B51 MOV 8DH, #0FAH
00B81 MOV 98H, #50H
00BB1 SETB 8EH
00BD1 MOV 0E0H, 2AH

```

DI: 7 LD: 10

AND Gate Disabled

Key Bounce Disabled

Standard

0.0 V output

Scope

DAC

BF 0 AC 0x00 IR 0x00 DR 0x00

8888

0.0 V input

11111111

ADC

MAX MIN

Motor Enabled

▲ After SemaphoreSignal (full)
(mutex=1, full=2, empty=1, _1_turn=0, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

Executed 0x00AE: INC 33H | Time: 9ms 31us - Inst

Assembly Code:

```

00931 MOV A, #03H
00951 CJNE A, 2DH, #03H
00981 MOV 2DH, #00H
009B1 MOV A, 2FH
009D1 INC A
009E1 MOV 2FH, A
00A01 MOV A, #0AH
00A21 CJNE A, 2FH, #03H
00A51 MOV 2FH, #00H
00A81 SETB 0AFH
00AA1 INC 29H
00AC1 INC 2AH
00AE1 INC 33H
00B01 SJMP 0B1H
00B21 ORL 89H, #20H
00B51 MOV 8DH, #0FAH
00B81 MOV 98H, #50H
00BB1 SETB 8EH
00BD1 MOV 0E0H, 2AH

```

DI: 7 LD: 10

AND Gate Disabled

Key Bounce Disabled

Standard

0.0 V output

Scope

DAC

BF 0 AC 0x00 IR 0x00 DR 0x00

8888

0.0 V input

11111111

ADC

MAX MIN

Motor Enabled

▲ After SemaphoreSignal (_1_turn)
(mutex=1, full=2, empty=1, _1_turn=1, _2_turn=0).

- Consumer:

System Clock (MHz): 11.0592 | 200 | Update Freq

8051

Copyright ©2005-2024 James Rogers

Remove All Breakpoints

Assembly Code:

```

00B0 SJMP 0B1H
00B2* ORL 89H, #20H
00B5 MOV 8DH, #0FAH
00B8 MOV 98H, #50H
00BB SETB 8EH
00BD MOV 0E0H, 2AH
00C0 JZ 0FBH
00C2 JB 0E7H, 0F8H
00C5 DEC 2AH
00C7 MOV 0E0H, 29H
00CA JZ 0FBH
00CC JB 0E7H, 0F8H
00CF DEC 29H
00D1 CLR 0AFH
00D3 MOV A, 2CH
00D5 ADD A, #30H
00D7 MOV R1, A
00D8 MOV 99H, @R1
00DA JBC 99H, 02H

```

I/O Window:

- DI: 7, LD
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 0.0 V output
- Scope
- DAC
- 8-bit UART @ 4800 Baud
- Rx Reset
- Tx Send
- ADC: 11111111
- Motor Enabled

▲ Just enter Consumer, after finish UART init
(mutex=1, full=2, empty=1, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq

8051

Copyright ©2005-2024 James Rogers

Remove All Breakpoints

Assembly Code:

```

00B0 SJMP 0B1H
00B2* ORL 89H, #20H
00B5 MOV 8DH, #0FAH
00B8 MOV 98H, #50H
00BB SETB 8EH
00BD MOV 0E0H, 2AH
00C0 JZ 0FBH
00C2 JB 0E7H, 0F8H
00C5 DEC 2AH
00C7 MOV 0E0H, 29H
00CA JZ 0FBH
00CC JB 0E7H, 0F8H
00CF DEC 29H
00D1 CLR 0AFH
00D3 MOV A, 2CH
00D5 ADD A, #30H
00D7 MOV R1, A
00D8 MOV 99H, @R1
00DA JBC 99H, 02H

```

I/O Window:

- DI: 7, LD
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 0.0 V output
- Scope
- DAC
- 8-bit UART @ 4800 Baud
- Rx Reset
- Tx Send
- ADC: 11111111
- Motor Enabled

▲ After SemaphoreWait (full)
(mutex=1, full=1, empty=1, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	0x00	B	0x00
0x00	0x00	0x02	0x1A	R6	0x00	ACC	0x01
RKD	THD			R5	0x00	PSW	0x19
1	1	TMOD	0x20	R4	0x00	IP	0x00
SCON	0x50	TCOM	0x50	R3	0x00	IE	0x82
				R2	0x00	PCON	0x00
pins	bits	TH1	TL1	R1	0x00	DPH	0x00
0xFF	0xFF	P3	0xFA	R0	0x00	DPL	0x00
0xFF	0xFF	P2	0x0E			SP	0x6F
0xFF	0xFF	P1					
0xFF	0xFF	P0					

PC: 8051

Modify RAM

addr	0x00	0x00	value
0	0	1	2
1	2	3	4
2	3	4	5
3	4	5	6
4	5	6	7
5	6	7	8
6	7	8	9
7	8	9	A
8	9	A	B
9	A	B	C
A	B	C	D
B	C	D	E
C	D	E	F
D	E	F	
E	F		
F			

Remove All Breakpoints

Copyright ©2005-2024 James Rogers

Assembly Code:

```

000CF: DEC 29H
000D0: SJMP 0B1H
000D1: ORL 89H,#20H
000D2: MOV 8DH,#0FAH
000D3: MOV 98H,#50H
000D4: SETB 8EH
000D5: MOV 0E0H,2AH
000D6: JZ 0FBH
000D7: JB 0E7H,0F8H
000D8: DEC 2AH
000D9: MOV 0E0H,29H
000DA: JZ 0FBH
000DB: JB 0E7H,0F8H
000DC: DEC 29H
000DD: CLR 0AFH
000DE: MOV A,2CH
000DF: ADD A,#30H
000E0: MOV R1,A
000E1: MOV 99H,@R1
000E2: JRC 99H,02H
  
```

Hardware Interface:

- DI: LD
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 8-bit UART @ 4800 Baud
- No Parity
- Rx Reset
- Tx Send
- 0.0 V input
- ADC
- Motor Enabled
- Scope DAC
- 8888

▲ After SemaphoreWait (mutex)

(mutex=0, full=1, empty=1, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	0x00	B	0x00
0x00	0x41	0x46	0x0E	R6	0x00	ACC	0x03
RKD	THD			R5	0x00	PSW	0x18
1	1	TMOD	0x20	R4	0x00	IP	0x00
SCON	0x50	TCOM	0x00	R3	0x00	IE	0x82
				R2	0x00	PCON	0x00
pins	bits	TH1	TL1	R1	0x30	DPH	0x00
0xFF	0xFF	P3	0xFC	R0	0x00	DPL	0x00
0xFF	0xFF	P2	0x0C			SP	0x6F
0xFF	0xFF	P1					
0xFF	0xFF	P0					

PC: 8051

Modify RAM

addr	0x00	0x00	value
0	0	1	2
1	2	3	4
2	3	4	5
3	4	5	6
4	5	6	7
5	6	7	8
6	7	8	9
7	8	9	A
8	9	A	B
9	A	B	C
A	B	C	D
B	C	D	E
C	D	E	F
D	E	F	
E	F		
F			

Remove All Breakpoints

Copyright ©2005-2024 James Rogers

Assembly Code:

```

000EE: INC 29H
000EF: SJMP 0B1H
000F0: MOV A,2CH
000F1: INC A
000F2: MOV 2CH,A
000F3: MOV A,#03H
000F4: CJNE A,2CH,03H
000F5: MOV 2CH,#00H
000F6: SETB 0AFH
000F7: INC 29H
000F8: INC 2BH
000F9: SJMP 0C9H
000FA: MOV 29H,#01H
000FB: MOV 2AH,#00H
000FC: MOV 2BH,#03H
000FD: MOV 33H,#01H
000FE: MOV 34H,#00H
000FF: MOV 2DH,#00H
00100: MOV 2CH,#00H
00101: MOV 2EH,#00H
00102: MOV 23H,#00H
  
```

Hardware Interface:

- DI: LD
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 8-bit UART @ 4800 Baud
- No Parity
- Rx Reset
- Tx Send
- 0.0 V input
- ADC
- Motor Enabled
- Scope DAC
- 8888

▲ After SemaphoreSignal (mutex)

(mutex=1, full=1, empty=1, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	B
0x00	0x41	0x46	0x0F	0x00	0x03

pins bits

TH1	TL1
0x00	0x00

PC: 8051

Data Memory

addr	0x00	0x00	value
0	0	1	2

Remove All Breakpoints

Assembly Code:

```

0000  SJMP 0001
0001  MOV A,2CH
0002  INC A
0003  MOV 2CH,A
0004  MOV A,#03H
0005  CJNE A,2CH,03H
0006  MOV 2CH,#00H
0007  SETB 0AFH
0008  INC 29H
0009  INC 2BH
000A  SJMP 0C9H
000B  MOV 29H,#01H
000C  MOV 2AH,#00H
000D  MOV 2BH,#03H
000E  MOV 33H,#01H
000F  MOV 34H,#00H
0010  MOV 2DH,#00H
0011  MOV 2CH,#00H
0012  MOV 2EH,#00H
0013  MOV 2FH,#00H
0014  MOV 2FH,#00H
0015  MOV 2FH,#00H
0016  MOV 2FH,#00H
0017  MOV 2FH,#00H
0018  MOV 2FH,#00H
0019  MOV 2FH,#00H
0020  MOV 2FH,#00H
0021  MOV 2FH,#00H
0022  MOV 2FH,#00H
0023  MOV 2FH,#00H
0024  MOV 2FH,#00H
0025  MOV 2FH,#00H
0026  MOV 2FH,#00H
0027  MOV 2FH,#00H
0028  MOV 2FH,#00H
0029  MOV 2FH,#00H
0030  MOV 2FH,#00H
0031  MOV 2FH,#00H
0032  MOV 2FH,#00H
0033  MOV 2FH,#00H
0034  MOV 2FH,#00H
0035  MOV 2FH,#00H
0036  MOV 2FH,#00H
0037  MOV 2FH,#00H
0038  MOV 2FH,#00H
0039  MOV 2FH,#00H
0040  MOV 2FH,#00H
0041  MOV 2FH,#00H
0042  MOV 2FH,#00H
0043  MOV 2FH,#00H
0044  MOV 2FH,#00H
0045  MOV 2FH,#00H
0046  MOV 2FH,#00H
0047  MOV 2FH,#00H
0048  MOV 2FH,#00H
0049  MOV 2FH,#00H
0050  MOV 2FH,#00H
0051  MOV 2FH,#00H
0052  MOV 2FH,#00H
0053  MOV 2FH,#00H
0054  MOV 2FH,#00H
0055  MOV 2FH,#00H
0056  MOV 2FH,#00H
0057  MOV 2FH,#00H
0058  MOV 2FH,#00H
0059  MOV 2FH,#00H
0060  MOV 2FH,#00H
0061  MOV 2FH,#00H
0062  MOV 2FH,#00H
0063  MOV 2FH,#00H
0064  MOV 2FH,#00H
0065  MOV 2FH,#00H
0066  MOV 2FH,#00H
0067  MOV 2FH,#00H
0068  MOV 2FH,#00H
0069  MOV 2FH,#00H
0070  MOV 2FH,#00H
0071  MOV 2FH,#00H
0072  MOV 2FH,#00H
0073  MOV 2FH,#00H
0074  MOV 2FH,#00H
0075  MOV 2FH,#00H
0076  MOV 2FH,#00H
0077  MOV 2FH,#00H
0078  MOV 2FH,#00H
0079  MOV 2FH,#00H
0080  MOV 2FH,#00H
0081  MOV 2FH,#00H
0082  MOV 2FH,#00H
0083  MOV 2FH,#00H
0084  MOV 2FH,#00H
0085  MOV 2FH,#00H
0086  MOV 2FH,#00H
0087  MOV 2FH,#00H
0088  MOV 2FH,#00H
0089  MOV 2FH,#00H
0090  MOV 2FH,#00H
0091  MOV 2FH,#00H
0092  MOV 2FH,#00H
0093  MOV 2FH,#00H
0094  MOV 2FH,#00H
0095  MOV 2FH,#00H
0096  MOV 2FH,#00H
0097  MOV 2FH,#00H
0098  MOV 2FH,#00H
0099  MOV 2FH,#00H

```

I/O Window:

- DI: 1, LD: 1
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 8-bit UART @ 4800 Baud
- Rx: No Parity
- Rx Reset
- Tx: Tx Send
- 0.0V output
- Scope
- DAC
- ADC: 11111111
- Motor Enabled

▲ After SemaphoreSignal (empty)
(mutex=1, full=1, empty=2, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	B
0x00	0x41	0x46	0x18	0x00	0x01

pins bits

TH1	TL1
0x00	0x00

PC: 8051

Data Memory

addr	0x00	0x00	value
0	0	1	2

Remove All Breakpoints

Assembly Code:

```

0000  SJMP 0001
0001  MOV A,2CH
0002  INC A
0003  MOV 2CH,A
0004  MOV A,#03H
0005  CJNE A,2CH,03H
0006  MOV 2CH,#00H
0007  SETB 0AFH
0008  INC 29H
0009  INC 2BH
000A  SJMP 0C9H
000B  MOV 29H,#01H
000C  MOV 2AH,#00H
000D  MOV 2BH,#03H
000E  MOV 33H,#01H
000F  MOV 34H,#00H
0010  MOV 2DH,#00H
0011  MOV 2CH,#00H
0012  MOV 2EH,#00H
0013  MOV 2FH,#00H
0014  MOV 2FH,#00H
0015  MOV 2FH,#00H
0016  MOV 2FH,#00H
0017  MOV 2FH,#00H
0018  MOV 2FH,#00H
0019  MOV 2FH,#00H
0020  MOV 2FH,#00H
0021  MOV 2FH,#00H
0022  MOV 2FH,#00H
0023  MOV 2FH,#00H
0024  MOV 2FH,#00H
0025  MOV 2FH,#00H
0026  MOV 2FH,#00H
0027  MOV 2FH,#00H
0028  MOV 2FH,#00H
0029  MOV 2FH,#00H
0030  MOV 2FH,#00H
0031  MOV 2FH,#00H
0032  MOV 2FH,#00H
0033  MOV 2FH,#00H
0034  MOV 2FH,#00H
0035  MOV 2FH,#00H
0036  MOV 2FH,#00H
0037  MOV 2FH,#00H
0038  MOV 2FH,#00H
0039  MOV 2FH,#00H
0040  MOV 2FH,#00H
0041  MOV 2FH,#00H
0042  MOV 2FH,#00H
0043  MOV 2FH,#00H
0044  MOV 2FH,#00H
0045  MOV 2FH,#00H
0046  MOV 2FH,#00H
0047  MOV 2FH,#00H
0048  MOV 2FH,#00H
0049  MOV 2FH,#00H
0050  MOV 2FH,#00H
0051  MOV 2FH,#00H
0052  MOV 2FH,#00H
0053  MOV 2FH,#00H
0054  MOV 2FH,#00H
0055  MOV 2FH,#00H
0056  MOV 2FH,#00H
0057  MOV 2FH,#00H
0058  MOV 2FH,#00H
0059  MOV 2FH,#00H
0060  MOV 2FH,#00H
0061  MOV 2FH,#00H
0062  MOV 2FH,#00H
0063  MOV 2FH,#00H
0064  MOV 2FH,#00H
0065  MOV 2FH,#00H
0066  MOV 2FH,#00H
0067  MOV 2FH,#00H
0068  MOV 2FH,#00H
0069  MOV 2FH,#00H
0070  MOV 2FH,#00H
0071  MOV 2FH,#00H
0072  MOV 2FH,#00H
0073  MOV 2FH,#00H
0074  MOV 2FH,#00H
0075  MOV 2FH,#00H
0076  MOV 2FH,#00H
0077  MOV 2FH,#00H
0078  MOV 2FH,#00H
0079  MOV 2FH,#00H
0080  MOV 2FH,#00H
0081  MOV 2FH,#00H
0082  MOV 2FH,#00H
0083  MOV 2FH,#00H
0084  MOV 2FH,#00H
0085  MOV 2FH,#00H
0086  MOV 2FH,#00H
0087  MOV 2FH,#00H
0088  MOV 2FH,#00H
0089  MOV 2FH,#00H
0090  MOV 2FH,#00H
0091  MOV 2FH,#00H
0092  MOV 2FH,#00H
0093  MOV 2FH,#00H
0094  MOV 2FH,#00H
0095  MOV 2FH,#00H
0096  MOV 2FH,#00H
0097  MOV 2FH,#00H
0098  MOV 2FH,#00H
0099  MOV 2FH,#00H

```

I/O Window:

- DI: 1, LD: 1
- AND Gate Disabled
- Key Bounce Disabled
- Standard
- 8-bit UART @ 4800 Baud
- Rx: No Parity
- Rx Reset
- Tx: Tx Send
- 0.0V output
- Scope
- DAC
- ADC: 11111111
- Motor Enabled

▲ After SemaphoreWait (full)
(mutex=1, full=0, empty=2, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

Executed 0x00CF: DEC 29H | Time: 20ms 263us - 1y

Assembly Code:

```

00AEI INC 33H
00B0I SJMP 0B1H
00B2I ORL 89H, #20H
00B5I MOV 8DH, #0FAH
00B8I MOV 98H, #50H
00BBI SETB 8EH
00BD* MOV 0E0H, 2AH
00C0I JZ 0FBH
00C2I JB 0E7H, 0F8H
00C5I DEC 2AH
00C7I MOV 0E0H, 29H
00CAI JZ 0FBH
00CC1 JB 0E7H, 0F8H
00CFI DEC 29H
00D1I CLR 0AFH
00D3I MOV A, 2CH
00D5I ADD A, #30H
00D7I MOV R1, A
00D8I MOV 99H, @R1
  
```

Copyright ©2005-2024 James Rogers

▲ After SemaphoreWait (mutex)
(mutex=0, full=0, empty=2, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592 | 200 | Update Freq.

Executed 0x00EE: INC 29H | Time: 22ms 329us - 1y

Assembly Code:

```

00D7I MOV R1, A
00D8I MOV 99H, @R1
00DAI JBC 99H, 02H
00DDI SJMP 0FBH
00DFI MOV A, 2CH
00E1I INC A
00E2I MOV 2CH, A
00E4I MOV A, #03H
00E6I CJNE A, 2CH, 03H
00E9I MOV 2CH, #00H
00ECI SETB 0AFH
00EE* INC 29H
00F0I INC 2BH
00F2I SJMP 0C9H
00F4I MOV 29H, #01H
00F7I MOV 2AH, #00H
00FAI MOV 2BH, #03H
00FDI MOV 33H, #01H
0100I MOV 34H, #00H
  
```

Copyright ©2005-2024 James Rogers

▲ After SemaphoreSignal (mutex)
(mutex=1, full=0, empty=2, _1_turn=1, _2_turn=0).

System Clock (MHz): 11.0592, 200 Update Freq.

RST Step Run New Load Save CPY Paste BP

Executed 0x00F0: INC 2BH | Time: 22ms 330us - I/O

Assembly Code:

```

00D7: MOV R1,A
00D8: MOV 99H,@R1
00DA: JBC 99H,02H
00DD: SJMP 0FBH
00DF: MOV A,2CH
00E1: INC A
00E2: MOV 2CH,A
00E4: MOV A,#03H
00E6: CJNE A,2CH,03H
00E9: MOV 2CH,#00H
00EC: SETB 0AFH
00EE: INC 29H
00F0: INC 2BH
00F2: SJMP 0C9H
00F4: MOV 29H,#01H
00F7: MOV 2AH,#00H
00FA: MOV 2BH,#03H
00FD: MOV 33H,#01H
0100: MOV 34H,#00H

```

Register Window:

R7	0x00	B	0x00
R6	0x00	ACC	0x03
R5	0x00	PSW	0x18
R4	0x00	IP	0x00
R3	0x00	IE	0x82
R2	0x00	PCON	0x00
R1	0x31	DPH	0x00
R0	0x00	DPL	0x00
SP	0x6F		

PC: 8051

Data Memory:

addr	0x00	0x00	value
00	23	21	07
01	22	23	00
02	23	00	00
03	00	00	00
04	00	00	00
05	00	00	00
06	00	00	00
07	00	00	00
08	00	00	00
09	00	00	00
0A	00	00	00
0B	00	00	00
0C	00	00	00
0D	00	00	00
0E	00	00	00
0F	00	00	00
10	00	00	00
11	00	00	00
12	00	00	00
13	00	00	00
14	00	00	00
15	00	00	00
16	00	00	00
17	00	00	00
18	00	00	00
19	00	00	00
1A	00	00	00
1B	00	00	00
1C	00	00	00
1D	00	00	00
1E	00	00	00
1F	00	00	00

I/O Window:

DI: 1, LD: 1

AND Gate Disabled

Key Bounce Disabled

Standard

0.0V output

Scope DAC

8-bit UART @ 4800 Baud

Rx: A, Tx:

Rx Reset, Tx Send

0.0V input

1111111

ADC

MAX, MIN, Motor Enabled

▲ After SemaphoreSignal (empty)
(mutex=1, full=0, empty=3, _1_turn=1, _2_turn=0).

- Unfair version:

System Clock (MHz): 11.0592, 200 Update Freq.

RST Step Pause New Load Save CPY Paste BP

Time: 52ms 700us - Instructions: 24600

Assembly Code:

```

ORG 0000H
0000: LJMP 010CH
0003: RETI
ORG 000BH
000B: LJMP 0113H
000E: LJMP 00DCH
0011: LJMP 000EH
0014: MOV 0E0H,2BH
0017: JZ 0FBH
0019: JB 0E7H,0F8H
001C: DEC 2BH
001E: MOV 0E0H,29H
0021: JZ 0FBH
0023: JB 0E7H,0F8H
0026: DEC 29H
0028: CLR 0AFH
002A: MOV A,2DH
002C: ADD A,#30H
002E: MOV R1,A

```

Register Window:

R7	0x01	B	0x00
R6	0x00	ACC	0x00
R5	0x00	PSW	0x18
R4	0x00	IP	0x00
R3	0x00	IE	0x82
R2	0x00	PCON	0x00
R1	0x32	DPH	0x00
R0	0x23	DPL	0x00
SP	0x6F		

PC: 8051

Data Memory:

addr	0x00	0x00	value
00	23	21	07
01	22	23	00
02	23	00	00
03	00	00	00
04	00	00	00
05	00	00	00
06	00	00	00
07	00	00	00
08	00	00	00
09	00	00	00
0A	00	00	00
0B	00	00	00
0C	00	00	00
0D	00	00	00
0E	00	00	00
0F	00	00	00
10	00	00	00
11	00	00	00
12	00	00	00
13	00	00	00
14	00	00	00
15	00	00	00
16	00	00	00
17	00	00	00
18	00	00	00
19	00	00	00
1A	00	00	00
1B	00	00	00
1C	00	00	00
1D	00	00	00
1E	00	00	00
1F	00	00	00

I/O Window:

DI: 1, LD: 1

AND Gate Disabled

Key Bounce Disabled

Standard

0.0V output

Scope DAC

8-bit UART @ 4800 Baud

Rx: ABCDEF, Tx:

Rx Reset, Tx Send

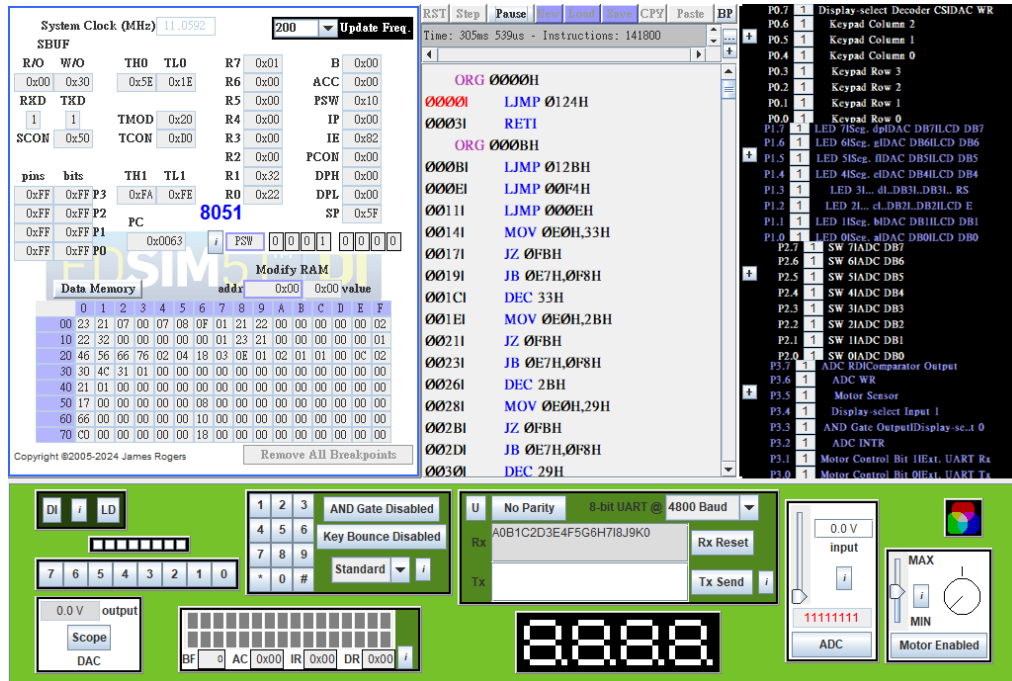
0.0V input

1111111

ADC

MAX, MIN, Motor Enabled

- Fair version:



Difference of them is that in fair version, I add additional semaphores “_1_turn” and “_2_turn” to indicate which producer should produce. For example, if it’s Producer1’s turn, then _1_turn=1 and _2_turn=0, and _2_turn will be signaled if and only if Producer1 finish, at the same time, _1_turn will be set to 0, which means it’s time for Producer2 to work.