Question	Answer
1	B
2	C
3	C
4	C
5	B
6	A
7	C
8	C
9	D
10	A
11	C
12	C
13	C
14	E
15	A
16	B
17	C
18	A
19	B
20	C
21	A
22	B
23	C
24	E
25	F
26	D
27	A
28	C
29	D
30	D
31	C
32	C
33	B

Test 1

Name_____

You are permitted 75 minutes to take this test. This is a closed book, closed notes test. You are allowed the following items for the test: calculators, pencils, pens, and erasers. You are not permitted to have a computer or other electronic assistance.

There are 35 questions. Each question is valued at 3 points each. You receive 1 point if your name is on all pages of the test. For a perfect score you need to answer 33 of the 37 questions. The 4 questions you are not answering MUST BE circled.

Please read and sign this statement: I have not received assistance from anyone nor assisted others while taking this test. I have also notified the test proctor of any violations of the above conditions.

Signature _____

Name _____

Table 23.2 Recommended Operating Conditions (1) (1)

Symbol	Parameter			Standard			Unit
				Min.	Typ.	Max.	
V_{CC1}, V_{CC2}	Supply Voltage($V_{CC1} \geq V_{CC2}$)			2.7	5.0	5.5	V
AVcc	Analog Supply Voltage				V_{CC1}		V
Vss	Supply Voltage				0		V
AVss	Analog Supply Voltage				0		V
V_{IH}	HIGH Input Voltage	P3_1 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7		0.8V_{CC2}		V_{CC2}	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (during single-chip mode)		0.8V_{CC2}		V_{CC2}	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (data input during memory expansion and microprocessor modes)		0.5V_{CC2}		V_{CC2}	V
		P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE		0.8V_{CC1}		V_{CC1}	V
		P7_0 , P7_1		0.8V_{CC1}		6.5	V
V_{IL}	LOW Input Voltage	P3_1 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7		0		0.2V_{CC2}	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (during single-chip mode)		0		0.2V_{CC2}	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (data input during memory expansion and microprocessor modes)		0		0.16V_{CC2}	V
		P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE		0		0.2V_{CC1}	V
$I_{OH (peak)}$	HIGH Peak Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1				-10.0	mA
$I_{OH (avg)}$	HIGH Average Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1				-5.0	mA
$I_{OL (peak)}$	LOW Peak Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1				10.0	mA
$I_{OL (avg)}$	LOW Average Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1				5.0	mA
f (XIN)	Main Clock Input Oscillation Frequency (4)		V_{CC1}=3.0 to 5.5V	0		16	MHz
			V_{CC1}=2.7 to 3.0V	0		20 X V_{CC1}-44	MHz
f (XCIN)	Sub-Clock Oscillation Frequency				32.768	50	kHz
f (Ring)	On-chip Oscillation Frequency			0.5	1	2	MHz
f (PLL)	PLL Clock Oscillation Frequency (4)		V_{CC1}=3.0 to 5.5V	10		24	MHz
			V_{CC1}=2.7 to 3.0V	10		46.67 X V_{CC1}-116	MHz
f (BCLK)	CPU Operation Clock			0		24	MHz
$t_{su(PLL)}$	PLL Frequency Synthesizer Stabilization Wait Time		V_{CC1}=5.0V			20	ms
			V_{CC1}=3.0V			50	ms

18. A/D Converter

The microcomputer contains one A/D converter circuit based on 10-bit successive approximation method configured with a capacitive-coupling amplifier. The analog inputs share the pins with P10_0 to P10_7, P9_5, P9_6, P0_0 to P0_7, and P2_0 to P2_7. Similarly, $\overline{\text{ADTRG}}$ input shares the pin with P9_7. Therefore, when using these inputs, make sure the corresponding port direction bits are set to "0" (= input mode).

When not using the A/D converter, set the VCUT bit to "0" (= Vref unconnected), so that no current will flow from the VREF pin into the resistor ladder, helping to reduce the power consumption of the chip.

The A/D conversion result is stored in the ADi register bits for ANi, AN0_i, and AN2_i pins (i = 0 to 7).

Table 18.1 shows the performance of the A/D converter. Figure 18.1 shows the block diagram of the A/D converter, and Figures 18.2 and 18.3 show the A/D converter-related registers.

Table 18.1 Performance of A/D Converter

Item	Performance
Method of A/D Conversion	Successive approximation (capacitive coupling amplifier)
Analog input Voltage (1)	0V to AVCC (VCC1)
Operating clock øAD (2)	fAD/divide-by-2 of fAD/divide-by-3 of fAD/divide-by-4 of fAD/divide-by-6 of fAD/divide-by-12 of fAD
Resolution	8-bit or 10-bit (selectable)
Integral Nonlinearity Error	When AVCC = VREF = 5V • With 8-bit resolution: ±2LSB • With 10-bit resolution AN0 to AN7 input, AN0_0 to AN0_7 input and AN2_0 to AN2_7 input : ±3LSB ANEX0 and ANEX1 input (including mode in which external Op-Amp is connected) : ±7LSB When AVCC = VREF = 3.3V • With 8-bit resolution: ±2LSB • With 10-bit resolution AN0 to AN7 input, AN0_0 to AN0_7 input and AN2_0 to AN2_7 input : ±5LSB ANEX0 and ANEX1 input (including mode in which external Op-Amp is connected) : ±7LSB
Operating Modes	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, and repeat sweep mode 1
Analog Input Pins	8 pins (AN0 to AN7) + 2 pins (ANEX0 and ANEX1) + 8 pins (AN0_0 to AN0_7) + 8 pins (AN2_0 to AN2_7)
A/D Conversion Start Condition	• Software trigger The ADST bit in the ADCON0 register is set to "1" (A/D conversion starts) • External trigger (retriggerable) Input on the $\overline{\text{ADTRG}}$ pin changes state from high to low after the ADST bit is set to "1" (A/D conversion starts)
Conversion Speed Per Pin	• Without sample and hold function 8-bit resolution: 49 øAD cycles, 10-bit resolution: 59 øAD cycles • With sample and hold function 8-bit resolution: 28 øAD cycles, 10-bit resolution: 33 øAD cycles

NOTES:

1. Does not depend on use of sample and hold function.
2. øAD frequency must be 12 MHz or less. And divide the fAD if VCC1 is less than 4.0V, and øAD frequency into 10 MHz or less.
   When sample & hold function is disabled, øAD frequency must be 250kHz or more.
   When sample & hold function is enabled, øAD frequency must be 1MHz or more.
3. If VCC2 < VCC1, do not use AN0_0 to AN0_7 and AN2_0 to AN2_7 as analog input pins.

Name _____

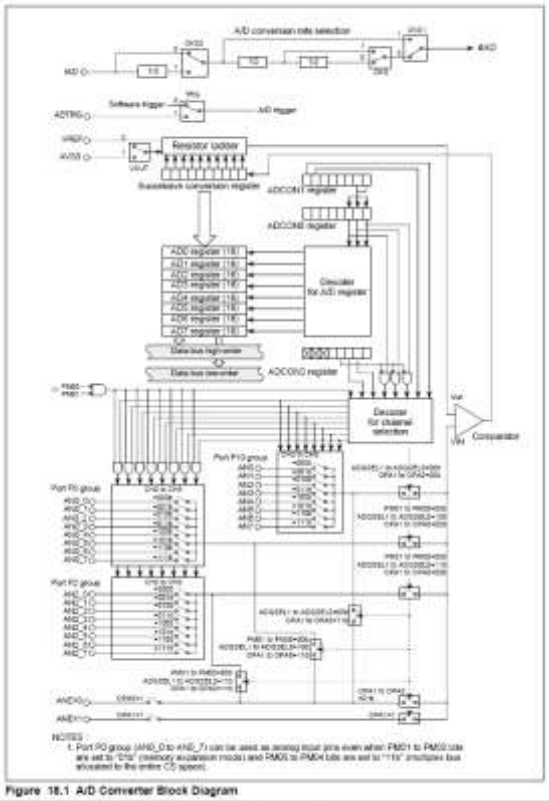


Figure 18.1 A/D Converter Block Diagram

2.1.2 Sections Managed by NC30

NC30 manages areas in which data and code are located as "sections".
This section explains the types of sections generated and managed by NC30 and how they are managed.

Sections Types

NC30 classifies data into sections by type for management purposes. (See Figure 2.1.3.)
Table 2.1.1 lists the sections types managed by NC30.

Table 2.1.1 Sections types Managed by NC30

Section base name	Content
data	Contains static variables with initial values.
bss	Contains static variables without initial values.
rom	Contains character strings and constants.
program	Contains programs.
vector	Variable vector area (compiler does not generate)
fvector	Fixed vector area (compiler does not generate)
stack	Stack area (compiler does not generate)
heap	Heap area (compiler does not generate)

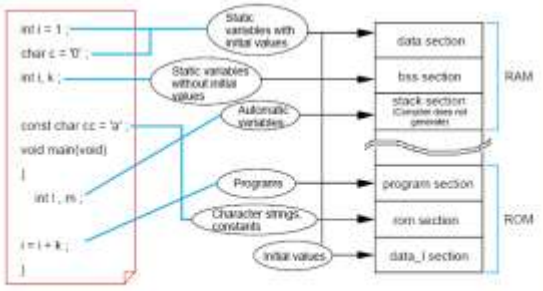


Figure 2.1.3 Mapping Data into Sections by Type

RENESAS M16C/60, M16C/20, M16C/Tiny, R8C/Tiny Series
Programming Guidelines <C Language>

Rules for Passing Arguments

NC30 has two methods for passing arguments to a function: "via a register" and "via a stack".
When the following three conditions are met, arguments are passed via a register; otherwise, arguments are passed via a stack.

(1) The types of the function's arguments are prototype declared.
(2) One or more arguments are the type that can be assigned to a register.
(3) No short-cut form is used in the argument part of prototype declaration.

Table 2.4.1 Rules for Passing Arguments

Type of argument	First argument	Second argument	Third and following arguments
char type	R1L	Stack	Stack
short, int types near pointer type	R1	R2	Stack
Other types	Stack	Stack	Stack

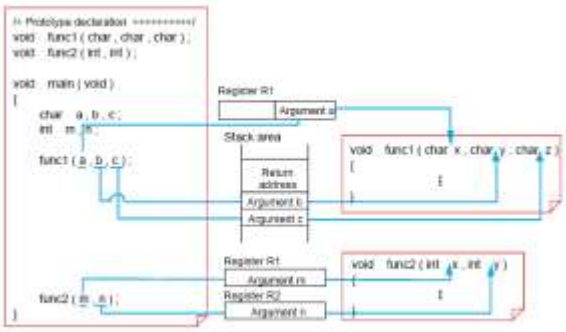


Figure 2.4.4 Example for Passing Arguments to Functions

Rules for Passing Return Values

All return values except those expressed by a struct or union, are stored in registers.
However, different registers are used to store the return values depending on their data types.
The return values represented by a struct or union are passed via "stored address and stack". Namely, an area to store a return value is prepared when calling a function, and this address is passed via a stack as a hidden argument. The called function writes its return value to the area indicated by the address placed in the stack when control returns from it.

Table 2.4.2 Rules for Passing Return Value

Data type	Returning method
char	R0L
int short	R0
long float	R2R0
double	R3R2R1R0
near pointer	R0
far pointer	R2R0
struct union	Store address is passed via a stack

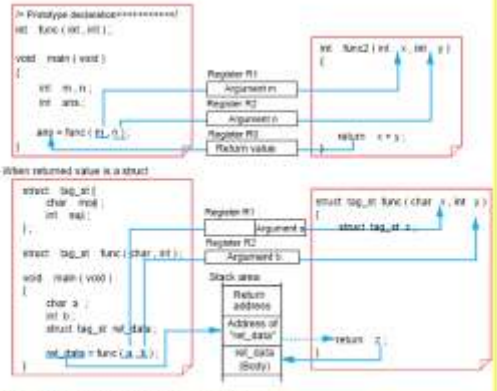
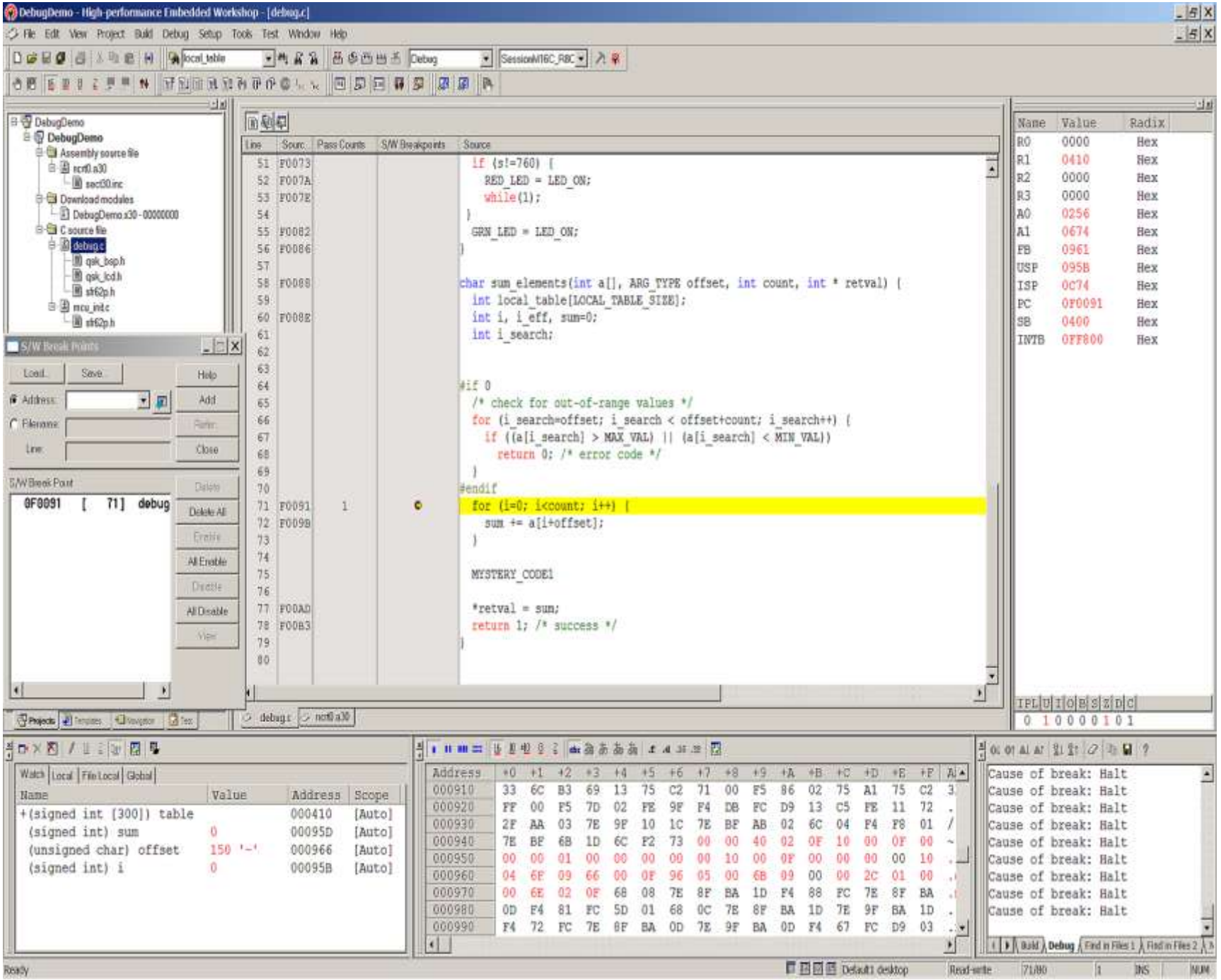


Figure 2.4.5 Example for Passing Return Value

Name _____


DebugDemo - High-performance Embedded Workshop - [debug.c]
File Edit View Project Build Debug Setup Tools Test Window Help
DebugDemo
Assembly source file
Download modules
C source file
debug.c
S/W Break Points
Load... Save... Help Add Close Delete Delete All Enable All Enable Disable All Disable View
0F0091 [71] debug
Line Source Pass Counts S/W Breakpoints Source
51 F0073 if (s!=760) {
52 F007A RED_LED = LED_ON;
53 F007E while(1);
54 }
55 F0082 GRN_LED = LED_ON;
56 F0086 }
57
58 F0088 char sum_elements(int a[], ARG_TYPE offset, int count, int * retval) {
59 int local_table[LOCAL_TABLE_SIZE];
60 F008E int i, i_eff, sum=0;
61 int i_search;
62
63
64 #if 0
65 /* check for out-of-range values */
66 for (i_search=offset; i_search < offset+count; i_search++) {
67 if ((a[i_search] > MAX_VAL) || (a[i_search] < MIN_VAL))
68 return 0; /* error code */
69 }
70 #endif
71 F0091 1 for (i=0; i<count; i++) {
72 F009B sum += a[i+offset];
73 }
74
75 MYSTERY_CODE1
76
77 F00AD *retval = sum;
78 F00B3 return 1; /* success */
79 }
80
Name Value Radix
R0 0000 Hex
R1 0410 Hex
R2 0000 Hex
R3 0000 Hex
A0 0256 Hex
A1 0674 Hex
FB 0961 Hex
USP 095B Hex
ISP 0C74 Hex
PC 0F0091 Hex
SB 0400 Hex
INTB 0FF800 Hex
IPL U I O B S Z D C
0 1 0 0 0 0 1 0 1
Watch Local File Local Global
Name Value Address Scope
+(signed int [300]) table 000410 [Auto]
(signed int) sum 0 00095D [Auto]
(unsigned char) offset 150 '~' 000966 [Auto]
(signed int) i 0 00095B [Auto]
Address +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F
000910 33 6C B3 69 13 75 C2 71 00 F5 86 02 75 A1 75 C2
000920 FF 00 F5 7D 02 FE 9F F4 DB FC D9 13 C5 FE 11 72
000930 2F AA 03 7E 9F 10 1C 7E BF AB 02 6C 04 F4 F8 01
000940 7E BF 6B 1D 6C F2 73 00 00 40 02 0F 10 00 0F 00
000950 00 00 01 00 00 00 00 00 10 00 0F 00 00 00 00 10
000960 04 6F 09 66 00 0F 96 05 00 6B 09 00 00 2C 01 00
000970 00 6E 02 0F 68 08 7E 8F BA 1D F4 88 FC 7E 8F BA
000980 0D F4 81 FC 5D 01 68 0C 7E 8F BA 1D 7E 9F BA 1D
000990 F4 72 FC 7E 8F BA 0D 7E 9F BA 0D F4 67 FC D9 03
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Cause of break: Halt
Build Debug Find in Files 1 Find in Files 2
Ready
Default desktop
Read-write
INS
NUM

Name _____

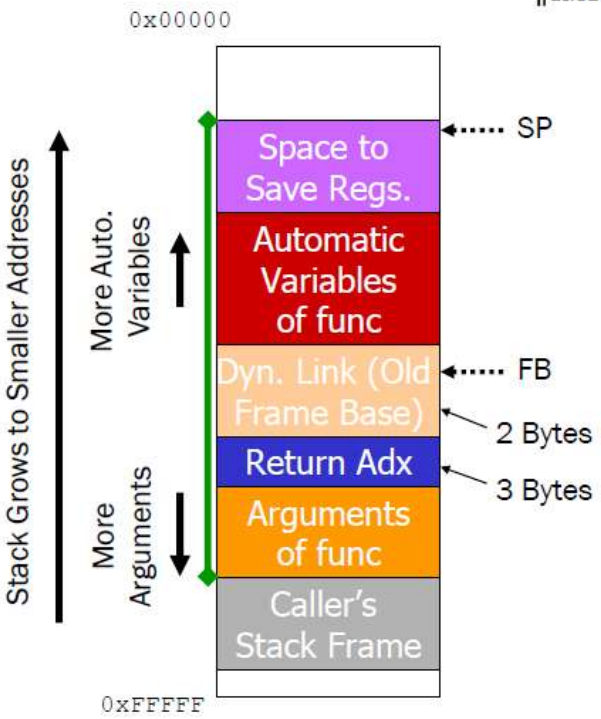
1. In the debug snapshot provided earlier, what is the return address?

   a. 0x6F0966
   b. 0x0F0066
   d. 0x0F6600
   e. 0x66000F

Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	A
000910	33	6C	B3	69	13	75	C2	71	00	F5	86	02	75	A1	75	C2	3
000920	FF	00	F5	7D	02	FE	9F	F4	DB	FC	D9	13	C5	FE	11	72	.
000930	2F	AA	03	7E	9F	10	1C	7E	BF	AB	02	6C	04	F4	F8	01	/
000940	7E	BF	6B	1D	6C	F2	73	00	00	40	02	0F	10	00	0F	00	~
000950	00	00	01	00	00	00	00	00	10	00	0F	00	00	00	00	10	.
000960	04	6F	09	66	00	0F	96	05	00	6B	09	00	00	2C	01	00	.
000970	00	6E	02	0F	68	08	7E	8F	BA	1D	F4	88	FC	7E	8F	BA	.
000980	0D	F4	81	FC	5D	01	68	0C	7E	8F	BA	1D	7E	9F	BA	1D	.
000990	F4	72	FC	7E	8F	BA	0D	7E	9F	BA	0D	F4	67	FC	D9	03	.

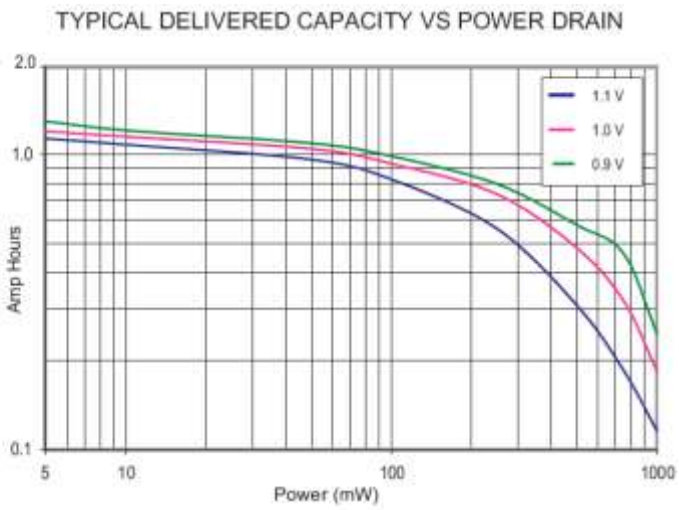
Name	Value	Radix
R0	0000	Hex
R1	0410	Hex
R2	0000	Hex
R3	0000	Hex
A0	0256	Hex
A1	0674	Hex
FB	0961	Hex
USP	095B	Hex
ISP	0C74	Hex
PC	0F0091	Hex
SB	0400	Hex
INTB	0FF800	Hex

- Frame base == *dynamic link*
- 5 bytes used for
  - old frame base (0[FB], 1[FB])
  - return address (2[FB], 3[FB], 4[FB])
- *enter* and *exitd* instructions used to
  - modify, save and restore SP and FB
  - return from subroutine



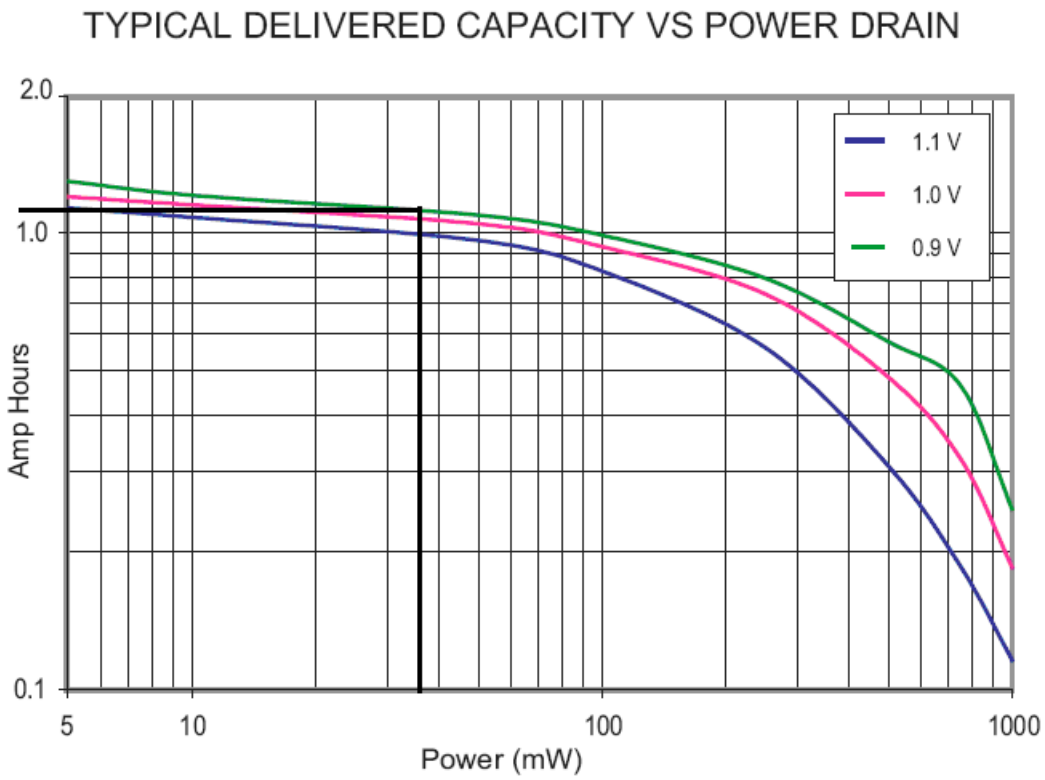
Name _____

Nominal Voltage:	1.5 V
Operating Voltage	1.6 - 0.75V
Impedance:	114 m-ohm @ 1kHz
Typical Weight:	11 gm (0.4 oz.)
Typical Volume:	3.5 cm^3 (0.2 $in.^3$)
Terminals:	Flat
Storage Temperature Range:	-20°C to 35°C
Operating Temperature Range:	-20°C to 54°C (-4°F to 130°F)
ANSI: **IEC:**	24A LR03



2. You have developed a toy pet rock using a different low power processor that contains several leds and noise makers and is operating with 4 "AAA" batteries. It is designed it to operate at 3.2v. This is well within the specifications for the system. Using a volt-amp meter the average current consumption is measured at 44.4mA. The marketing department has determined the average play time for the toy is 10 minutes per day. How many days will the batteries last if it operated for 10 minutes a day?

a. ~1 month
b. ~2 month
c. ~4 month
d. ~8 month



Name _____

The first item to determine is the power consumed.

44.4 mA * 3.2 v 142.08 mW

142.08 mW / 4 batteries = 35.52 mW per battery

Using the chart for the battery and moving up to the 0.9v line [4 batteries x 0.9v = 3.6v]. Moving up the 35.5mW line on the x-axis and over to the y-axis, the available current is 1.0+ Amp-Hour. Having 4 batteries quadruples the voltage but keeps the current constant at 1.0+ Amp-Hour operating for 10 minutes a day, consumes

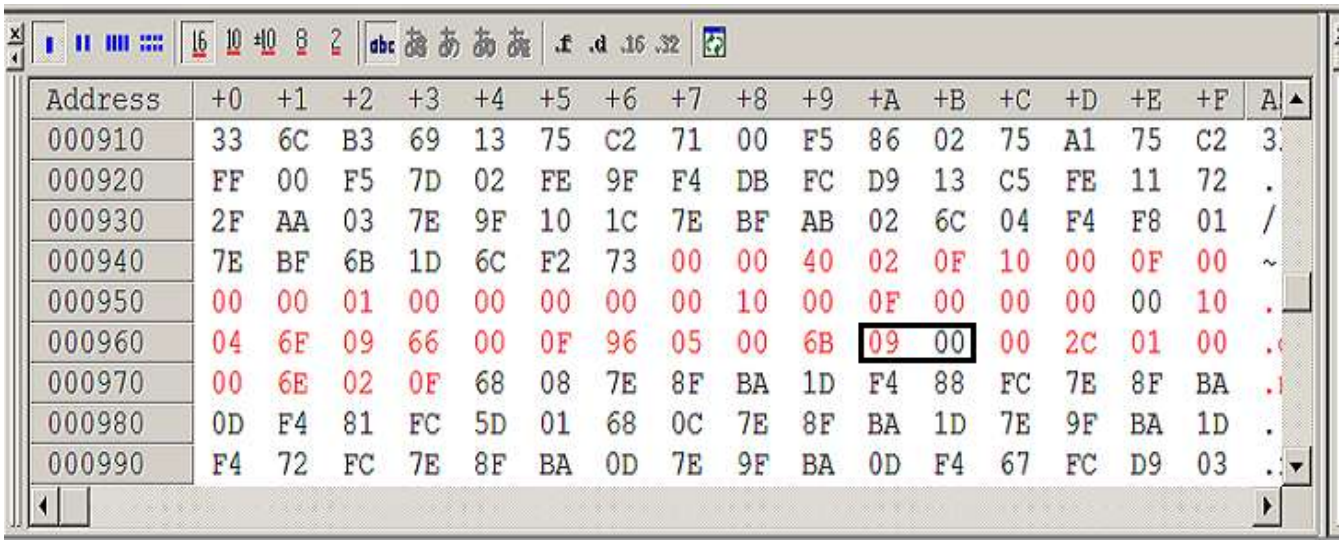
$$10 \text{ min} * \frac{1 \text{ hour}}{60 \text{ min}} * 44.4 \text{ mA} \quad 7.4 \text{ mA-H}$$

1000 mAmp-Hour / 7.4 mA-H ==> 135 days

a. ~1 month
b. ~2 month
c. ~4 month
d. ~8 month

3. With the Byte aligned system, in the course of debugging your 12 bit Analog to Digital converter, [refer to debug screen provided] you have the result stored in an unsigned integer at location 0x00096A. What is the value stored?

a. 0x096B
b. 0x6B09
c. 0x0009
d. 0x90B6



Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	A
000910	33	6C	B3	69	13	75	C2	71	00	F5	86	02	75	A1	75	C2	3
000920	FF	00	F5	7D	02	FE	9F	F4	DB	FC	D9	13	C5	FE	11	72	.
000930	2F	AA	03	7E	9F	10	1C	7E	BF	AB	02	6C	04	F4	F8	01	/
000940	7E	BF	6B	1D	6C	F2	73	00	00	40	02	0F	10	00	0F	00	~
000950	00	00	01	00	00	00	00	00	10	00	0F	00	00	00	00	10	.
000960	04	6F	09	66	00	0F	96	05	00	6B	09	00	00	2C	01	00	.
000970	00	6E	02	0F	68	08	7E	8F	BA	1D	F4	88	FC	7E	8F	BA	.
000980	0D	F4	81	FC	5D	01	68	0C	7E	8F	BA	1D	7E	9F	BA	1D	.
000990	F4	72	FC	7E	8F	BA	0D	7E	9F	BA	0D	F4	67	FC	D9	03	.

Name _____

4. Which of the following memory sections have known, predictable data when the microcontroller powers up (comes out of reset)?

   a. rom, program, bss
   b. bss, stack, heap
   c. rom, data
   d. data , stack, data_l

For the next eight questions, consider the following C code. Identify in which section(s) each variable or argument has space allocated.

```
int v1[4];
char v2='#';
char v6[10] = "project"
int function1(char v3) {
   char * v4;
   static float v5=1.4;
   v4 = &v6[0];
}
```

5. Identify the location of v1:

   a. data and data_l
   b. bss
   c. rom
   d. stack

6. Identify the location of v2:

   a. data and data_l
   b. bss
   c. rom
   d. stack

7. Identify the location of v3:

   a. data and data_l
   b. bss
   c. register
   d. stack and bss

8. What does v4 equal:

   a. "p"
   b. "project"
   c. memory address where the "p" in project is stored
   d. address of the stack pointer

9. How many bytes long is v1?

   a. 2
   b. 16
   c. 4
   d. 8

10. How many bytes long is v2?

    a. 1
    b. 4
    c. 2
    d. 8

11. How many bytes long is v4?

    a. 1
    c. 2

Name _____

b. 4
d. 8

12. How many locations in v6 are defined:

a. 9
b. 7
c. 8
d. 10

Given the following function call prototype, identify through which storage mechanism each argument is held when passed to the function.

```
void fun_1(char arg1, float arg2, int arg3);
```

13. How is arg1 passed?

a. R0L
b. R0H
c. R1L
d. R1H
e. Stack
f. Heap

14. How is arg2 passed?

a. R0
b. R1
c. R2
d. R2R0
e. Stack
f. Heap

15. Given the following function call prototype where is the return value held after the function completes?

```
char Calc_Offset(float arg1, int arg2, int arg3);
```

a. R0L
b. R0H
c. R0
d. R1
e. Stack
f. Heap

Name _____

Consider the following code. What are the contents as the code executes? (? indicates unknown characters)

```
          void main (void) {
            int i, j;
            int *p1, *p2;
1           i = 8;
2           j = 3;
3           p1 = &i;
4           p2 = &j;
5           *p1 = *p1+*p2;
6           p2 = p1;
          }
```

			Lines of code					
	Address	Variables	1	2	3	4	5	6
	0x000924	i	0x08	0x08	0x08	0x08	?[19]	
	0x000925		0x00	0x00	0x00	0x00		
	0x000926	j		0x03	0x03	0x03		
	0x000927			0x00	0x00	0x00		
	0x000928	p1			?[17]			
	0x000929							
	?[16]	p2				?[18]		?[20]

16. What is the missing address?

a. 0x000930
b. 0x00092A
c. 0x000931
d. 0x00092B

17. What is the portion of p1 assigned in line 3 stored in address location 0x000928?

a. 0x25
b. 0x26
c. 0x24
d. 0x00

18. What is the portion of p2 assigned in line 4 stored in address location identified above?

a. 0x26
b. 0x24
c. 0x2A
d. 0x00

19. What is the portion of "i" assigned in line 5?

a. 0x11
b. 0x0B
c. 0x03
d. 0x08

20. What is the portion of p2 assigned in line 6?

a. 0x25
b. 0x26
c. 0x24
d. 0x22

Name _____

21. In a byte aligned system, consider the following function. How large is its activation record?

```
#define KNOWLEDGE 0X05
int bar(int a, char b) {
   char x, y[5], z[5];
   int idontknow;
   for (idontknow=0, idontknow <= KNOWLEDGE, idontknow++){
     z[idontknow] = y[idontknow];
}
```

a. 21
b. 12
c. 13
d. 14
e. 16
f. None of the above

+5 Frame Base / Return Address

+1 char x

+5 char y[5]

+5 char z[5]

+1 char b – passed value

+2 int idontoknow

+2 int a – passed value

21

22. Where does an M16C ISA processor's SB register point?

a. Beginning of heap
b. Beginning of static data area
c. Dynamic link in current activation record
d. Return address in current activation record
e. Top of stack
f. Next instruction to execute

23. Where does an M16C ISA processor's FB register point?

a. Beginning of heap
b. Beginning of static data area
c. Dynamic link in current activation record
d. Return address in current activation record
e. Top of stack
f. Next instruction to execute

24. Where does an M16C ISA processor's SP register point?

a. Beginning of heap
b. Beginning of static data area
c. Dynamic link in current activation record
d. Return address in current activation record
e. Top of stack
f. Next instruction to execute

25. Where does an M16C ISA processor's PC register point?

a. Beginning of heap
b. Beginning of static data area
c. Dynamic link in current activation record
d. Return address in current activation record
e. Top of stack
f. Next instruction to execute

26. Consider the following C array. Assume it begins at address 0x03E8. At which Memory address is element a[1][2] stored?

```
int a[4][5];
```

Name _____

a. 0x0412
b. 0x03FC
d. 0x03F6
e. 0x0400

0x03E8	A[0,0]	0x03EA	A[0,1]	0x03EC	A[0,2]	0x03EE	A[0,3]	0x03F0	A[0,4]
0x03F2	A[1,0]	0x03F4	A[1,1]	0x03F6	A[1,2]	0x03F8		0x03FA	
0x03FC	A[2,0]	0x03FE	A[2,1]	0x0400	A[2,2]				
	A[3,0]								

27. In the previous question, which byte is stored at that address?

a. LSB
b. MSB
d. The only byte
e. BSS

$$n = \left\lfloor \frac{(V_{in} - V_{-ref})2^N}{V_{+ref} - V_{-ref}} + 1/2 \right\rfloor$$

28. What is the output code (in hexadecimal) of an 10-bit ADC with

Vcc = 3.3v V_{in}=1.7 V, V_{+ref}=2.5 V, V_{-ref}= 0.5 V?

a. 0x0B30
b. 0x02B8
c. 0x0266
d. 0x0210

$$n = \left\lfloor \frac{(V_{in} - V_{-ref})2^N}{V_{+ref} - V_{-ref}} + 1/2 \right\rfloor$$

Vin	=	1.7
Vref+	=	2.0
Vref-	=	1.0
N	=	8

```
n = [ ( ( Vin - Vref- ) * ( 2 * * N ) ] / ( Vref+ - Vref- ) ) + 0.50
= [ ( ( 1.7 - 0.5 ) * ( 2 * * 10 ) ] / ( 2.5 - 0.5 ) ) + 0.50
= [ ( 1.2 * 1024 ) / 2 ] ) + 0.50
= [ 1228.8 / 2 ] + 0.50
= 614.4 + 0.50
= 614.9
= 614
= 0x0266
```

29. How many Analog to Digital Conversion units are in the M30260 MCU on your QSK62P?

a. 16
b. 8
c. 26
d. 1
e. 24
f. 2

30. How many bits of precision are provided by the ADC used in the M30260 MCU on your QSK62A?

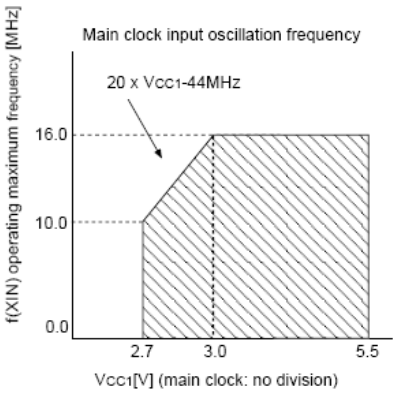
Name _____

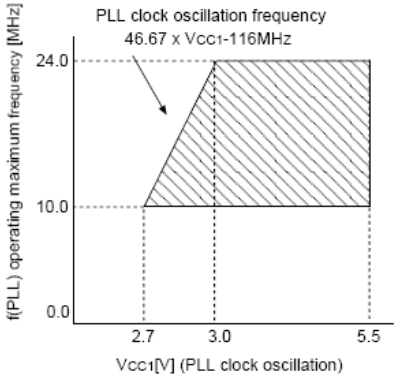
a. 8
b. 10
c. 12
d. 8 or 10
e. 10 or 12
f. 12 or 16

31. As an alarm notification, your design includes a red LED. The LED requires at least 1.6v to operate. It will illuminate with as little as 1mA. It maximum brightness is reached at 20mA. Your code will flash it at a rate just under 50% duty cycle. What is the maximum current can the QSK62P can provide directly?

a. 2mA
b. 5mA
c 10mA
d. 20mA

NOTES:
1. Referenced to $V_{CC1} = V_{CC2} = 2.7$ to 5.5V at T_{opr} = -20 to 85 °C / -40 to 85 °C unless otherwise specified.
2. The mean output current is the mean value within 100ms.
3. The total $I_{OL\ (peak)}$ for ports P0, P1, P2, P8_6, P8_7, P9, P10, P11, P14_0 and P14_1 must be 80mA max. The total $I_{OL\ (peak)}$ for ports P3, P4, P5, P6, P7, P8_0 to P8_4, P12, and P13 must be 80mA max. The total $I_{OH\ (peak)}$ for ports P0, P1, and P2 must be -40mA max. The total $I_{OH\ (peak)}$ for ports P3, P4, P5, P12, and P13 must be -40mA max. The total $I_{OH\ (peak)}$ for ports P6, P7, and P8_0 to P8_4 must be -40mA max. The total $I_{OH\ (peak)}$ for ports P8_6, P8_7, P9, P10, P11, P14_0, and P14_1 must be -40mA max.
4. Relationship between main clock oscillation frequency, PLL clock oscillation frequency and supply voltage.





5. There is no external connections for port P1_0 to P1_7, P4_4 to P4_7, P7_2 to P7_5 and P9_1 in 80-pin version.

32. If you power the QSK62P with 3 "AAA" batteries in series and you expect each battery will discharge to 0.9v. You have a connection of the raw battery voltage through a voltage divider to an analog-to-digital input and can monitor the health of the batteries. You can adjust how fast the processor runs based on the measurement read. What can you set your main clock frequency to?

a. 8Mhz
b. 10Mhz
c 16Mhz
d. 24Mhz

33. What is the minimum number of "AAA" batteries that are needed to power the QSK62P?

a. 1
b. 2
c 3
d. 4