How to use uBUG12

uBUG12 is a GUI to interface with Freescale's Serial Monitor that are preprogrammed into the NC12s and Adapt9S12E128 families. It has some similarities with Gordon Doughman's DBUG12.

uBUG12 can be downloaded from Technological Arts website <u>http://support.technologicalarts.ca/files/uBug12.zip</u>

For PCs with Windows98SE the .net framework must be installed in order for uBUG12 to run. WinXP, 2K the .net framework is (usually) already installed. The .net framework can be found at MS website

http://www.microsoft.com/downloads/details.aspx?FamilyID=d7158dee-a83f-4e21-b05a-009d06457787&displaylang=en

Getting Started:

Double click on the uBUG12 icon to start GUI. Below is what uBUG12 started.



for S12 Serial Monitor

Here uBUG12 is waiting for commands. By typing *help* one can see different commands that can be use.

ma uBug12	
File Help	
MonStatus ErrorText ComPort	

Type the *help* command

ma uBug12	- O X
File Help	
help	
·	
MonStatus ErrorText ComPort	11.

Once the help command is invoked, uBUG12 will list the different commands as shown.

💏 uBug12	_ 🗆 🗙
File Help	
>help	
REGISTER RD - Register Display RM <registername> <data8 16=""> - Register Modify CCR <data8> - Set CCR register D <data16> - Set PC register PC <data16> - Set PC register PY <data16> - Set P register X <data16> - Set X register Y <data16> - Set Y register MEMORY MODIFY BF <startadd> <endadd> <data8> - Block fill byte BFW <startadd> <endadd> <data16> - Block fill word MD <startadd> <endadd> - Memory display MDW <startadd> [<endadd>] - Memory display word MM <address> <data8> - Memory modify byte MMW <address> <data8> - Memory modify word FLASH FBULK - Flash bulk erase FLOAD [;B][;M] - Flash load GO/HALT DEVICE - Get device name GO/HALT CO[<startaddress] -="" execution<br="" start="">HALT - Halt execution RESET - Reset target TRACE - Execute one instruction BR <address16> <ppage> - Set breakpoint at specified address GUI CON <comport> - Connect to target DISCON - Disconnect from target EXIT - Terminate GUI HELP - Display help OP <opacity%> - Set main gui opacity</opacity%></comport></ppage></address16></startaddress]></data8></address></data8></address></endadd></startadd></endadd></startadd></data16></endadd></startadd></data8></endadd></startadd></data16></data16></data16></data16></data16></data8></data8></registername>	
No Connection Unknown Error ComPort	li.

The different commands will be explained throughout this document. This document will use the NC12 module with a Docking module to help explain the command types.

Connecting:

This document will use COM 1 of the PC to connect to the target as an example. For PC without serial port, a USB to COM can be purchase from any computer store.

Connect a Serial cable from COM 1 to the Docking Module. Slide the Run/Load switch to Load or Boot then power up the board. Make sure the power LED is on.

The command to connect is **CON 1** for COM 1 and **CON 2** for COM 2.

i 🖉 u	Bug12	
File	Help	
I 1		
I 1		
I 1		
I 1		
I 1		
I 1		
con	1	
P	Status ErrorText ComPort	1.

2 possible errors can occur:

Connection Error: Unable to open COM1 <- Another application is using the COM port

Connection Error: Read Error: Timeout error <- The MCU not currently in LOAD mode or the cable is disconnected from either PC or Docking Module, lastly the serial cable is connected on the wrong COM port.

ma uBug12	
File Help	
>con 1 Connection Error: Unable to open COM1 >con 1	
Connection Error: Read Error: Timeout error >con 1 CONNECTED	
Cold Reset Executed Unknown Error COM 1	

A **CONNECTED** message will appear to show good connection between PC and the target.

The *device* command will show the target type is as MC9S12C32 Rev 1.0.

🙀 uBug12	<u>_ U ×</u>
File Help	
>con 1 Connection Error: Unable to open COM1 >con 1 Connection Error: Read Error: Timeout error >con 1 CONNECTED >device MC9S12C32 Rev 1.0	
Monitor Active No Error COM 1	1.

Disconnecting:

To disconnect uBUG12 from the serial port, the command **discon**. This would allow other application to use the COM 1 like HyperTerm or Tera Term.

🙀 uBug12	
File Help	
>con 1 CONNECTED >discon DISCONNECTED	
Cold Reset Executed Unknown Error COM 1	li.

Memory Dump:

The commands are *MD* and *MDW*. For example, type *md 4000 4100* to dump an address block.

💏 uBug12	
File Help	
>con 1 Connection Error: Unable to open COM1 >con 1 Connection Error: Read Error: Timeout error >con 1 CONNECTED >device MC9512C32 Rev 1.0	
md 4000 4100	
Monitor Active No Error COM 1	li.

If for whatever reason the cable is disconnected, power disruption, MCU is reset the error may appear ">md 4000 4100 Connection error: Read Error: Timeout error". Reconnect again by typing CON 1.

The *MD* command is byte dump.

wag12	
File Help	
<pre>>con 1 Connection Error: Unable to open COM1 >con 1 Connection Error: Read Error: Timeout error >con 1 CONNECTED >device MC9S12C32 Rev 1.0 >md 4000 4100 Connection error: Read Error: Timeout error >con 1 CONNECTED >md 4000 4100</pre>	
40E0 - FF FF FF FF - FF FF FF FF - FF FF FF	
4110 - FF FF FF FF - FF FF FF - FF FF FF - FF FF	
Monitor Active No Error COM 1	1.

The **MDW** command is word dump.

i i i i i i i i i i i i i i i i i i i	lug 1	2																_	x
File	Help																		
4100 4110 >mdw 4000 4010 4020 4030 4040 4050 4050	40	FF FFF FFF FFF FFF FFF FFF	FF 410 FF FF FF FF FF FF	FFF FFFF FFFF FFFF FFFF FFFF FFFF	FF 	- FF +4 FFFF FFFF FFFF FFFF FFFF FFFF	FF +6 FFF FFF FFF FFF FFF FFF	F	FF - +8 FFFI FFFI FFFI FFFI FFFI FFFI	FFFF FFFF FFFF FFFF FFFF FFFF	Ff - - - - -	FFF FFFF FFFF FFFF FFFF FFFF FFFF	- FI FFI FFI FFI FFI FFI FFI	F FF FF FF FF FF FF FF FF	 	 			 •
4070 4080 4090 40A0 40B0 40C0 40C0 40E0 40E0		FFF FFF FFF FFF FFF FFF FFF	= F = F = F = F = F	FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF		FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF		F - F - F - F - F - F -	FFFI FFFI FFFI FFFI FFFI FFFI FFFI	FFFF FFFF FFFF FFFF FFFF FFFF		FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF	FFI FFI FFI FFI FFI	FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF					
Monito	or Ac	tive	-	No Err	or	COM	1												

Memory Fill:

The commands are *BF* and *BFW*. Example of byte fill, type *bf* 3800 3900 AA to fill an address block with the value \$AA

i 🖉 ul	Bug1	2																							
File	Help																								
4100) –	FF	FF	FF	FF	-	FF	FF	FF	FF	-	FF	FF	FF	FF	-	FF	FF	FF	FF					
4110) · · - ·	FF	FF	FF	FF	_	FF	FF	FF	FF	_	FF	FF	FF	FF	_	FF	FF	FF	FF					
>mdv	· 400	 DO 4	10	 0																					
1.000		+0		+2			4	+ 6			+8		+A		+C		_+E								
4000		FFF	-	FFFF FFFF			FF	FFF	•		FFF FFF		FFF FFF		FFF		FFFF		:::	:::	:::	:::	:::	::	
4020		FFF	-	FFFF FFFF			FF	FFF			FFF FFF		FFF FFF		FFF FFF	-	FFFF		• • •	• • •		• •	• • •	• •	
4040) –		-	FFFF			FF				FFF		FFF		FFF		FFFF		:::		:::	::		::	
4050		FFF	-	FFFF			FF	FFF	-	-	FFF		FFF		FFF	-	FFFF		• • •	• • •		• •	• • •	• •	
4070	<u>i –</u>	FFF	F	FFFF																	•••	•••			
bf 3							<u></u>	-																	
Monit	or Ac	tive	1	lo Err	or	ΠC	ОМ	1																	11.

i gu	Bug 1	2											
File	Help												
4070 4080 4090 4000 4000 4000 4000 4000 50f) -) -) -) -) -	FFFF FFFF FFFF FFFF FFFF FFFF	FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF 0 aa		FFFF FFFF FFFF FFFF FFFF FFFF	FFFF FFFF FFFF FFFF FFFF FFFF	 FFFF FFFF FFFF FFFF FFFF FFFF	FFFF FFFF FFFF FFFF FFFF FFFF	 FFFF FFFF FFFF FFFF FFFF FFFF	FFFF FFFF FFFF FFFF FFFF FFFF			
Monit	or Ac	tive	No Erro	or	COM	1							

Verifying the block fill

💏 uBug12	
File Help	
4090 - FFFF FFFF - FFFF - FFFF - FFFF - FFFF FFFF 40A0 - FFFF FFFF - FFFF - FFFF FFFF - FFFF FFFF	
md 3800 3900 Monitor Active No Error COM 1	

Memory dump of the address block \$3800 to \$3900

🙀 ul	3ug 1	2																							_	
File	Help)																								
40E0			FF I												FFFF	FI	FFFF	-								
40F0			FF I		F —	FF	FFF	FFF	FF ·	- FF	FFI	F FI	FFF	- 1	FFFF	FΙ	FFFF	-								
			900	aa																						
>md	380																									
			+1						+6					+A					+E							
3800			AA						AA		-			AA		-			AA						 	
3810			AA						AA						AA				AA						 	
3820			AA						AA						AA				AA						 	
3830			AA						AA		-				AA	-			AA						 	
3840		AA	AA	AA	AA	-			AA		-			AA		-			AA						 	
3850			AA						AA						AA		AA	AA	AA	AA					 	
3860			AA						AA						AA				AA						 	
3870			AA						AA						AA		AA	AA	AA	AA					 	
3880			AA						AA						AA				AA						 	
3890			AA						AA						AA		AA	AA	AA	AA					 	
38A0			AA						AA						AA		AA	AA	AA	AA					 	. —
38BO			AA				AA		AA					AA					AA						 	
38C0			AA						AA						AA		AA								 	
3800		AA	AA	AA	AA	-			AA		-				AA	-	AA	AA	AA	AA					 	
38E0			AA				AA		AA		-			AA		-			AA						 	
38F0		AA	AA	AA	AA		AA		AA		-	AA	AA		AA	-		AA	AA	AA					 	
3900		AA		25		-	4F	5C	95		-	4D	08	80	23	-	C2	03	01	84						
3910		23	12	81	8F	-	40	в8	1E	57	-	E7	14	⊂7	1F	-	2A	06	59	4E	#.	@	۶۱	Ν	 *.YI	N
Monit	or Ac	tive		lo Er	ror		:OM	1																		

Example of word fill, type *bfw 3800 3900 5555* to fill an address block with the value \$5555

Sig ut	Bug 1	2																										_ [[1×
File	Help																												
40E0) –	FFI	FF	FFF		FF	FF	FFI	F -	- FF	FFF	F FF	FFF	- 1	FFFF	: 1	FFFF	- ,											
40F0) –	FFI	FF	FFFF		FF	FFF	FFI	FF -	- FF	FFF	FFF	FFF	- I	FFFF	- 1	FFFF	Ξ.											
>bf			900	aa																									
>md	3800																												
				+2					+6					+A				+D											
3800		AA		AA	AA	-			AA		-	AA			AA	-		AA			• •	•••	••	••	••	••	••	• • •	
3810		AA				-	AA		AA		-	AA			AA	-		AA			• •	•••	••	••	••	••	••	• • •	
3820		AA.	AA.	AA.	AA.	-	AA	AA	AA		-	AA	AA	AA	AA.	-	AA.			AA .	• •	• •	••	••	••	••	••	• • •	
3830		AA				-	AA		AA		-	AA				-		AA			• •	•••	••	••	••	••	••	• • •	
3840		AA	AA			-	AA		AA	AA	-	AA	AA	AA		-	AA				• •	•••	••	••	•••	••	••	• • •	
3850		AA	AA AA			-	AA AA		AA		-	AA AA		AA		-		AA AA			• •	•••	•••	••	•••	•••	••	• • •	
3860		AA					AA		AA						AA	-		AA			• •	•••	•••	•••	•••	•••	••	• • •	
3870		AA AA			AA AA		AA AA		AA			AA			AA AA	Ξ	AA AA				• •	•••	•••	•••	•••	•••	••	• • •	_
						_	AA	АА	AA	AA	_	AA	AA	AA	AA	_	AA	AA	AA	AA			•••	•••	•••	•••	<u></u>		
bfw	3800	0 3:	900	555	55																								
Monit	or Acl	tive	N	lo Er	ror	C	ЮM	1																					1.

) jiji u	Bug1	2																				_ 🗆	×
File	Help																						
38A	J —	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA			
38B) –	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA			
380) –	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA			
38D	- c	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA	-	AA	AA	AA	AA			
38E					AA									AA									
38F		AA			AA			AA										AA					
390	_	AA	87	25	OC.	_	4F		95		_	4D	08		23		C2			84	%.O\M#		
391	- c	23	12	81	8F	_	40				_	É7	14	Ĉ7	1F			06			#@W		
	- N 381	ກດ້າ	3900	1 5	555																	• • • •	
1 2 1																							-
						_					_					_							_
Moni	tor Ac	tive	N	lo Er	ror	0	:OM	1															1.

Memory dump of the address block \$3800 to \$3900

i ja ul	3ug 1	2												
File	Help)												
>mdw	/ 38	00 390	00											
		+0	+2		+4	+6		+8	+A		+C	+E		
3800	· -	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3810	· -	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3820	· -	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3830		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3840		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3850	· -	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3860		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3870		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3880		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3890		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
38A0		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
38B0		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
38C0		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
38D0		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
38E0		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
38F0	- 1	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	-
Monit	or Ac	tive	No Erro	or	COM	1								1.

Memory Modify:

The commands are *MM* and *MMW*. Example of byte modify, type *mm 3800 AA* to modify the value at address memory \$3800 with \$AA.

i ge ul	3ug 1	2												
File	Help													
>mdv	/ 380	00 39	00											
		+0	+2		+4	+6		+8	+A		+C	+E		
3800	· -	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3810	- ו	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3820	- ו	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3830	- ו	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3840	· -	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3850		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3860		5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3870		5555	5555	-	5555	5555		5555	5555	-	5555	5555	000000000000000000000000000000000000000	
3880	- 1	5555	5555	-	5555	5555	-	5555	5555	-	5555	5555	000000000000000000000000000000000000000	_
mm 3	800	aa												
Monit	or Ac	tive	No Erro)r	COM	1								1.

i i i i i i i i i i i i i i i i i i i	3ug 1	2									
File	Help	1									
3870 3880 3890 38A0 38C0 38C0 38C0 38C0 38F0 >mm		5555 5555 5555 5555 5555 5555 5555 5555	5555 5555 5555 5555 5555		5555 5555 5555 5555 5555 5555 5555		 5555 5555 5555 5555 5555 5555 5555	5555 5555 5555 5555 5555	 5555 5555 5555 5555 5555 5555 5555 5555 5555	5555 5555 5555 5555 5555 5555 5555	•
 Monite	or Ac	tive	No Erro)(COM	1					

Verifying the memory modify. Note the byte change at \$3800.

ma uBug12	
File Help	
3880 - 5555 5555 - 5555 5555 000000000000000000000000000000000000	A
>mm 3800 aa >md 3800 +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 3800 - AA 55 55 55 - 55 55 55 - 55 55 55 55 55 55	
Monitor Active No Error COM 1	

Example of word modify, type *mmw 3800 FFFF* to modify the value at address memory \$3800 with \$FFFF.

ma uBug12	_ 🗆 ×
File Help	
38C0 - 5555 5555 - 5555 5555 - 5555 5555 - 5555 5555 UUUUUUUU	A
+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 3800 - AA 55 55 55 - 55 55 55 - 55 55 55 55 55 55	UUU •
Monitor Active No Error COM 1	

Verifying the memory modify. Note the word change at \$3800.

We uBug12	- U ×
File Help	
38F0 - 5555 5555 - 5555 5555 - 5555 5555 - 5555 5555 00000000	1
+0 +2 +4 +6 +8 +A +C +E 3800 - FFFF 5555 - 5555 5555 - 5555 5555 - 5555 55550000000000	•

Registers:

There are 9 commands to modify the different registers.

RD - Register Display RM <RegisterName> <Data8/16> - Register Modify CCR <Data8> - Set CCR register D <Data16> - Set D register PC <Data16> - Set PC register PP <Data8> - Set PP register SP <Data16> - Set SP register X <Data16> - Set X register Y <Data16> - Set Y register

Data8 – 8 bit value Data16 – 16 bit value

The *RD* command shows the content of PPAGE (**PP**), Program Counter (**PC**), Stack Pointer (SP), Conditional Code (**CCR**) **X**, **Y**, and **D** registers.

💏 uBug12	
File Help	
>con 1 CONNECTED >rd PP PC SP X Y D = A:B CCR = SXHI NZVC 00 C000 4000 0000 00:00 1101 0000	
Monitor Active No Error COM 1	1.

The *RM* command will allow the registers to be changed. For example, change Register X content with value 2000. Type the command *RM X 2000.*

Note that Register X has changed from \$0000 to \$2000

💏 uBug12	- O ×
File Help	
>con 1 CONNECTED >rd	
PP PC SP X Y D = A:B CCR = SXHI NZVC 00 C000 4000 0000 0000 00:00 1101 0000	
>rm × 2000 >rd	
PP PC SP X Y D = A:B CCR = SXHI NZVC	
00 C000 4000 2000 0000 00:00 1101 0000	
Monitor Active No Error COM 1	1

Notice the registers as each one are changed.

🍖 uBug12						
File Help						
>con 1 CONNECTED >rd)					
PP PC 00 C000 >rm × 200 >rd	4000			D = A:B 00:00	CCR = SXHI NZ 1101 00	
PP PC 00 C000 >ccr 0			Y 0000	D = A:B 00:00	CCR = SXHI NZ 1101 00	
		× 2000		D = A:B 00:00	CCR = SXHI NZ 0000 000	
PP PC 3F C000 >pc 4000	SP 4000	× 2000	Y 0000	D = A:B 00:00	CCR = SXHI NZ 0000 000	
			Y 0000	D = A:B 00:00	CCR = SXHI NZ 0000 00	
PP PC	SP 3900	× 5FD2		D = A:B 7F:55	$CCR = S \times HI NZ^{2}$ $0111 003$	
PP PC 3F A552 >y aaaa				D = A:B 7F:55	$CCR = S \times HI NZ'$ $0111 00:$	
PP PC 3F A552 >rd		× 5555		D = A:B 7F:55	$CCR = S \times HI NZ'$ $0111 00:$	
PP PC 3F A552			Y AAAA		CCR = S×HI NZ 0111 003	
				_		
Monitor Activ	∕e jNo	Error	COM 1			li.

Flash erase and programming:

To erase the Flash memory the command is **FBULK**.

🙀 uBug12	
File Help	
>con 1 CONNECTED	
fbulk	
Monitor Active Unknown Error COM 1	1.

Successful erase

🙀 uBug12	- D ×
File Help	
>con 1 CONNECTED >fbulk	
Monitor Active No Error COM 1	1.

To program FLASH the command is *Fload ;b* for banked S19, SX, S2 records. For non banked S2 or formatted S19 (went thru SrecCVT) record the command is *Fload*.

🙀 uBug12	- O ×
File Help	
>con 1 CONNECTED >fbulk	
fload ;b	
Monitor Active No Error COM 1	1.

Once the Fload ;b command is invoked, uBUG12 will open an explorer window to help and locate the S-record. In this example, demo.sx will be the target S-record file.

Double (click c	on the	file to	initiate	upload.
----------	---------	--------	---------	----------	---------

Fload Banked					<u>?</u> ×
Look <u>i</u> n:	C V102		•	+ 🗈 💣 🎟+	
My Recent Documents Desktop My Documents My Computer	Demo.SX				
S					
My Network Places	File <u>n</u> ame:	Demo.SX		~	<u>O</u> pen
	Files of <u>type</u> :	S19, S2, Sx Records		•	Cancel

Successful programming of Demo.sx

🐅 uBug12	
File Help	
CONNECTED >fbulk >fload ;b LOADED OKAY: 0.53125Sec. Tranfer rate was 7.5294Kb/sec	•
Monitor Active No Error COM 1	li.

Note that the Serial Monitor resides at \$F800 - \$FFFF. Therefore uBUG12 will automatically re-locate the vector addresses at below \$F800.

Briefly look at the Pseudo Vector address to check where the start of the program. The command is *md f7ff* to show a memory dump of the Pseudo Vector address at power up or reset.

The power up reset value at \$F7FE is \$C000. Therefore the program will start at \$C000

💏 uBug12	< I
File Help	
>con 1 CONNECTED >fbulk >fload ;b LOADED OKAY: 0.531255ec. Tranfer rate was 7.5294Kb/sec >md f7ff +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F F7F0 - C1 8F C0 00 - C0 00 C0 00 - C0 00 C0 00	
	-
Monitor Active No Error COM 1	/.

Memory dump at \$C000

🦐 uBug12	. D X
File Help	
>con 1 CONNECTED >fbulk >fload ;b LOADED OKAY: 0.53125Sec. Tranfer rate was 7.5294Kb/sec	
<pre>>md f7ff +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F F7F0 - C1 8F C0 00 - C0 00 C0 00 - C0 00 C0 00 - C0 00 C0 00 >md c000 +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F C000 - 14 10 18 0B - 00 00 11 18 - 0B 39 00 10 - CF 3F 80 799?.y</pre>	
Monitor Active No Error COM 1	

One can see that there are Data at \$C000. To execute the program using uBUG12, several registers needs to be initialized. Firstly, look at the registers by the *RD* command then invoke the *RESET* command to initialize the registers.

Memory dumps of the registers before **RESET** command is invoked.

💏 uBug12	
File Help	
>con 1 CONNECTED >fbulk >fload ;b LOADED OKAY: 0.53125Sec. Tranfer rate was 7.5294Kb/sec	
<pre>>md f7ff +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F F7F0 - C1 8F C0 00 - C0 00 C0 00 - C0 00 C0 00 - C0 00 C0 00 >md c000 +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F C000 - 14 10 18 0B - 00 00 11 18 - 0B 39 00 10 - CF 3F 80 79999</pre>	
Monitor Active No Error COM 1	

After **RESET** command please note the changes with various registers.

ma uBug12	
File Help	
LOADED OKAY: 0.53125Sec. Tranfer rate was 7.5294Kb/sec	
>md f7ff	
+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F F7F0 - C1 8F C0 00 - C0 00 C0 00 - C0 00 C0 00 - C0 00 C0 00	
+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F C000 - 14 10 18 0B - 00 00 11 18 - 0B 39 00 10 - CF 3F 80 79999	
PP PC SP X Y D = A:B CCR = SXHI NZVC 3F A552 3900 5555 AAAA 7F:55 0111 0011 >reset	
>rd PP PC SP X Y D = A:B CCR = SXHI NZVC 00 C000 4000 0000 0000 00:00 1101 0000	-
Monitor Active No Error COM 1	

To execute program the command is simply type **go** (after RESET is invoked) or **go C000**.

```
      File
      Help

      >md
      f7ff

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

      F7F0
      - C1
      8F
      C0
      00
      - C0
      00
      C0
      - C0
      00
      - C0
      00
      - C0
      - C
```

The BR, TRACE commands are not to be use and will at some point be fully implemented.

This concludes the uBUG12. Stay tune for further development.

Lastly, the **op** command will change the look of the GUI. One should try to see what it does as it cannot be shown in this document.