



Arduino Pro Primer

Day 3: Debugging Arduino Sketches

Sponsored by



MARAN.





DigiKey

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click "Help" or submit a question asking for assistance.
- Participate in 'Attendee Chat' by maximizing the chat widget in your dock.



DigiKey



Fred Eady

Visit 'Lecturer Profile' in your console for more details.



AGENDA





Build a Microchip 24CW160 EEPROM Driver Sketch
 Debug the Microchip 24CW160 EEPROM Driver





24CW160 Driver Variables and Constants

1	<pre>#include <wire.h></wire.h></pre>
2	
3	// I2C address for 24CW160 EEPROM
4	<pre>// Datasheet calls out 7-bit address = 0b1010000</pre>
5	<pre>// The final address to send is 0b01010000</pre>
6	<pre>const byte EEPROM_ADDR = 0x50;</pre>
7	
8	<pre>char writeBuf[64] = {"Microchip 24CW160 EEPROM Driver"};</pre>
9	<pre>char readBuf[64];</pre>
10	
11	<pre>// page size is 32 bytes for 24CW160 EEPROM</pre>
12	<pre>// page count is 64 pages for 24CW160 EEPROM</pre>
13	<pre>const unsigned int PAGE_SIZE = 32;</pre>
14	<pre>const unsigned int PAGE_NUM = 64;</pre>
15	
16	unsigned int startPage;
17	unsigned int endPage;
18	unsigned int offset;
19	unsigned int numberofpages;
20	unsigned int paddrposition;
21	unsigned int MemAddress;
22	unsigned int bytesremaining;
23	unsigned int data_indx;





24CW160 Driver *setup()* and *loop()*



25	<pre>void setup() {</pre>
26	// RX0/TX0
27	<pre>Serial1.begin(115200);</pre>
28	// Wirel = SDA1 & SCL1
29	Wirel.begin();
30	Wirel.setClock(400000);
31	EEPROM_Write(0,0,writeBuf,strlen(writeBuf
32	delay(5);
33	<pre>EEPROM_Read(0,0,readBuf,strlen(writeBuf))</pre>
34	<pre>Serial1.println(readBuf);</pre>
35	}
36	
37	<pre>void loop() {</pre>
38	}





DigiKey

24CW160 Driver - EEPROM_Write Function

sssss = 0x00-0x1F bytes per page (maximum 32 bytes per page)
pppppp = 0x00-0x3F pages (maximum 40 pages)

53	<pre>53 void EEPROM_Write(unsigned int page, unsigned int offset, char *data, unsigned int size)</pre>										
54 55	{ // calculate the beginning bit of the page addressing bits (paddrposition)										
56	<pre>// p = page addressing bits</pre>	addressing bits TABLE 3-3: FIRST WORD ADDRESS BYTE									
57	// s = page size bits	Memory Region	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
SQ	// AIO A9 AO A/ AO AS A4 AS A2 AI AO			2	2	2				2	
59	// pppppssss	16-Kbit EEPROM	0	x	x	x	x	A10	A9	A8	
60	paddrposition = 0x05;	32-Kbit EEPROM	0	x	x	x	A11	A10	A9	A8	
62	<pre>// calculate the start page and the end page</pre>	64-Kbit EEPROM	0	x	x	A12	A11	A10	A9	A8	
63	<pre>startPage = page;</pre>	128-Kbit EEPROM	0	x	A13	A12	A11	A10	A9	A8	
64	<pre>endPage = page +((size + offset)/PAGE_SIZE);</pre>	Configuration Registers	1	x	x	x	x	x	x	x	
65											
66	<pre>// number of pages to be written</pre>										
67	<pre>numberofpages = (endPage-startPage) + 1;</pre>	TABLE 3-4: SECOND W	IORD ADI	DRESS B	YTE						
68	<pre>// set writeBuf array index to 0x00</pre>	Memory Region	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
69	data_indx = 0x00;		۸7	A6	Λ <u>Γ</u>	A.4	A2	A2	Δ1	A0	
			AI	AO	AD	A4	AS	AZ		AU	
		32-Kbit EEPROM	A7	A6	A5	A4	A3	A2	A1	A0	

A7

A7

x

A6

A6

x

A5

A5

x

A4

A4

x

A3

A3

х

A2

A2

x

64-Kbit EEPROM

128-Kbit EEPROM

Configuration Registers⁽¹⁾

A0

A0

х

A1

A1

x



94

Arduino Pro Primer Debugging Arduino Sketches Build a Microchip 24CW160 EEPROM Driver Sketch

24CW160 Driver - EEPROM_Write Function







// read the data from EEPROM
for(int i=0; i<numberofpages; i++)</pre>

Arduino Pro Primer Debugging Arduino Sketches Build a Microchip 24CW160 EEPROM Driver Sketch

DigiKey

24CW160 Driver - EEPROM_Read Function

FIGURE 7-3:

114115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137

// add the page address to the byte address to // calculalte the beginning memory location MemAddress = startPage << paddrposition | offset;</pre> // calculate remaining number of bytes to be readn bytesremaining = bytesleft(size,offset); Wire1.beginTransmission(EEPROM ADDR); Wirel.write(highByte(MemAddress)); Wirel.write(lowByte(MemAddress)); Wirel.endTransmission(); Wirel.requestFrom(EEPROM ADDR, bytesremaining); while(Wire1.available()) readBuf[data indx++] = Wire1.read(); startPage += 1; // increment the page size = size - bytesremaining; // recalculate size of the data

1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9 1 2 3 4 5 7 8 9 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 SCL Device Address Byte Word Address – Byte 0 ----Word Address – Byte 1 ----SDA Start by ACK ACK ACK from Client Host from from Device Address Byte — — Data Byte (n) Data Byte (n+1) 1 \ 0 \ 1 \ 0 \ A2\A1\A0\ 1 \ 0 \ D7\D6\D5\D4\D3\D2\D1\D0\ 0 \ D7\D6\D5\D4\D3\D2\D1\D0\ 0 \ 0 \ D7\D6\D5\D4\D3\D2\D1\D0\ 0 \ (___∭MSb Start by ACK from Client ACK from Host ACK from Host 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9 -______ -Device Byte (n+2)-Data Byte (n+3)-Data Byte (n+x) {D7\D6\D5\D4\D3\D2\D1\D0\{ 0 \}D7\D6\D5\D4\D3\D2\D1\D0\{ 0 \}\D7\D6\D5\D4\D3\D2\D1\D0\{ 0 \}\D7\D6\D5\D4\D3\D2\D1\D0\{ 1 \} Stop by Host ACK from Host ACK from NACK Host The A13, A12 and A11 word address bits are "don't care" bits on the 24CW16X. Note 1: 2: The A13 and A12 word address bits are "don't care" bits on the 24CW32X.

3: The A13 word address bit is a "don't care" bit on the 24CW64X.

SEQUENTIAL READ



Verify the 24CW160 Driver







Output

/home/fred/.arduino15/packages/arduino/hardware/mbed_giga/4.1.3/bootloaders/GIGA/bootloader.elf syntax error: no colon char on the first line character at line 1

Using library Wire in folder: /home/fred/.arduino15/packages/arduino/hardware/mbed_giga/4.1.3/libraries/Wire (legacy) /home/fred/.arduino15/packages/arduino/tools/arm-none-eabi-gcc/7-2017q4/bin/arm-none-eabi-size -A /tmp/arduino/sketches/530D04C0491F47E86D0AE93F1255957D/24CW160_i2c.ino.elf Sketch uses 122192 bytes (6%) of program storage space. Maximum is 1966080 bytes. Global variables use 51976 bytes (9%) of dynamic memory, leaving 471648 bytes for local variables. Maximum is 523624 bytes.



Wire It All Up







Wire It All Up







Wire It All Up – Attach the J-Link

DigiKey



13



Fire Up the Ozone Debugger

	N	ew Project Wizard ×
Target Device Choose a Ta	arget Device	
Device		
STM32H747XI_I	M7	
Register Set		
Cortex-M7 (with	FPU)	
Derinherele (entie	201	
Periprierais (opilo	nai)	
Flash Banks		
Base Address	Name	Loader
0x0800 0000	Internal program flash	Default
0x0810 0000	Internal program flash	Default
0x9000 0000	External QSPI flash	CLK@PB2_nCS@PG6_D0@PF8_D1@PF9_D2@PF7_D3@P
<u> </u>		
		Cancel < Back Next >





Fire Up the Ozone Debugger

	New Proje	ct Wizard		×
Connection Settings Choose a Target and Host Interfa	ce			
arget Interface		Target Inter	ace Speed	
SWD	\$	4 MHz		~
ost Interface		Serial No (o	ptional)	
JSB	\$			
mulators connected via USB				
Product	Nicknam	ie	Serial No	
SEGGER J-Link ARM Pro			174301702	
		×	<u>C</u> ancel < <u>B</u> ack	<u>N</u> ext >





Fire Up the Ozone Debugger

New Project Wizard	×
Program File	
Choose the Program to be debugged	
ELF, Motorola S-record, Intel Hex, or Binary file (optional)	
/tmp/arduino/sketches/530D04C0491F47E86D0AE93F1255957D/24CW160_i2c.ino.elf	

Output

/home/fred/.arduino15/packages/arduino/hardware/mbed_giga/4.1.3/bootloaders/GIGA/bootloader.elf syntax error: no colon char on the first line character at line 1

Using library Wire in folder: /home/fred/.arduino15/packages/arduino/hardware/mbed_giga/4.1.3/libraries/Wire (legacy)

/home/fred/.arduino15/packages/arduino/tools/arm-none-eabi-gcc/7-2017q4/bin/arm-none-eabi-size -A /tmp/arduino/sketches/530D04C0491F47E86D0AE93F1255957D/24CW160_i2c.ino.elf Sketch uses 122192 bytes (6%) of program storage space. Maximum is 1966080 bytes.

Global variables use 51976 bytes (9%) of dynamic memory, leaving 471648 bytes for local variables. Maximum is 523624 bytes.





Fire Up the Ozone Debugger

New Project Wizard	×
Optional Settings Set optional project settings, such as the initial PC	
Initial PC (after download and reset)	
ELF Entry Point	
O Read from Base Address Vector Table	
O Read from Location	
 Location 	
Do not set	
Initial Stack Pointer	
Read from Base Address Vector Table	
Read from Location	
O Location	
O not set	
J-Link Script File	
J-Link Log File	
	Eack



Ready.

Arduino Pro Primer Debugging Arduino Sketches Debug the Microchip 24CW160 EEPROM Driver





18

Ln 0 Ch 0 Disconnected.

– 🗇 X Ozone - The J-Link Debugger V3.34 - *New Project View Find Debug Tools Window Help U - II (e - 3 + Functions × / main.cpp × Disassembly Name #include <Arduino.h> 20 ‡ *f* m (anonymous namespace)::pool::allocate 21 #include <USB/PluggableUSBSerial.h> License along with this library; if not, write to the Free Software 16 (anonymous namespace)::pool::free Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA 17 Unwind_Backtrace 22 18 */ Unwind ForcedUnwind 19 // Declared weak in Arduino.h to allow user redefinitions. 23 Unwind Resume or Rethrow 20 #include <Arduino.h> 21 #include <USB/PluggableUSBSerial.h> int atexit(void (* /*func*/)()) { return 0; } aeabi atexit 24 22 aeabi idiv0 23 // Declared weak in Arduino.h to allow user redefinitions. 25 aeabi Idivmod 24 int atexit(void (* /*func*/)()) { return 0; } // Weak empty variant initialization function. aeabi uldivmod 26 25 // Weak empty variant initialization function. 26 _aeabi_unwind_cpp_pr0 // May be redefined by variant files. 27 27 // May be redefined by variant files. aeabi unwind cpp pr1 28 void initVariant() __attribute__((weak)); void initVariant() attribute ((weak)); 28 aeabi unwind cpp pr2 29 void initVariant() { } _ascii_mbtowc 30 void initVariant() { } 29 __ascii_wctomb 31 void setupUSB() attribute ((weak)): 30 32 void setupUSB() { } _cxa_allocate_exception 33 _cxa_begin_catch void setupUSB() attribute ((weak)); 31 34 int main(void) cxa_begin_cleanup 35 ± { void setupUSB() { } 32 cxa call terminate 36 ± init(); cxa call unexpected 37 + initVariant(); 33 cxa end catch 38 34 39 #if defined(SERIAL CDC) int main(void) __cxa_end_cleanup _cxa_free_exception 35 ± { 41 + _SerialUSB.begin(115200); _cxa_get_globals 42 #endif 36 + init(); cxa get globals fast 43 cxa guard abort 44 + setup(); 37 + initVariant(); 45 __cxa_guard_acquire 46 for (;;) { 38 __cxa_guard_release 47 ± loop(); __cxa_init_primary_exception #if defined(SERIAL CDC) 39 if (arduino::serialEventRun) arduino::serialEventRun(); 48 + __cxa_pure_virtual() 49 } PluggableUSBD().begin(); 40 + cxa rethrow 50 51 __cxa_throw return 0; 41 + SerialUSB.begin(115200); 52 cya tyne match 53 42 #endif Memory 1 @ 00000000 43 File path resolved: "\$(InstallDir)/Config/CPU/Cortex-M7F.svd" was found at "/opt/SEGGER/Ozone V334/Config/CPU/Cortex-M7F.svd" 00000000 File.Open ("/tmp/arduino/sketches/530D04C0491F47E86D0AE93F1255957D/24CW160_i2c.ino.elf"); setup(); ·· · • 44 🕂 00000010 File.Open: completed in 103 ms 00000020 Program segments: 00000030 45 Address Size Code RO Data RW Data ZI Data Flg 00000040 for (;;) { 46 00000050 0805CE64 Θ Θ R 00000060 loop(); 47 + 08040000 118 372 117 480 892 0 RE 00000070 24000000 4 712 Θ 4 712 Θ RW 00000080 if (arduino::serialEventRun) arduino::serialEventRun(); 30000000 278 528 278 528 RW Θ 000000000 3800000 64 512 64 512 Θ RW 000000000 49 24001268 24 24 Θ RW 00000B0 50 24001280 47 264 47 264 RW 00000000 000000000 51 return 0; Total: 513 412 117 480 892 347 776 47 264 000000F0 52 For further information on ELF file data sections, execute command Elf.PrintSectionInfo(0) 00000100 Debug.ReadIntoInstCache: updated instruction information within 1 code ranges (0x08040000-0x0805CAE8) 00000110 00000120 00000130



Find the Sketch Source





Display the Sketch Source



	Ozone - The J-Lir	nk Debugger V3.34 - *New Project	24CW	/160 i2c.ino × V main.cpp × V			
Eile View Find Debug Tools Window	w <u>H</u> elp						
Ů▼ II 1€ - 3+ ± ±		📄 File	Scope				
Functions	\times 24CW160 i2c.ing \times main.cpp \times						
Name			1	#include <wire.h></wire.h>			
(anonymous namespace)::pool::allocate		↓ f EEPROM_Read	2				
(anonymous namespace)::pool::free	1 #include <wire.h></wire.h>		3	// T2C address for 24CW160 EEPROM			
Unwind Backtrace	2 J J T2C address for 24CW160 EEPBOM		5				
Unwind ForcedUnwind	4 // Datasheet calls out 7-bit address = 0b1010000		4	<pre>// Datasheet calls out 7-bit address = 0b1010000</pre>			
Unwind Resume or Rethrow	5 // The final address to send is 0b01010000		5	// The final address to send is 0b01010000			
_aeabi_atexit	<pre>6 const byte EEPROM_ADDR = 0x50;</pre>		6	const byto EERDOM ADDR - 0x50.			
aeabi_idiv0	7		0	const byte EERON_ADDK = 0x50;			
aeabi_ldivmod	char writeBul[64] = { microchip 240w100 EEPROM Driver };		7				
aeabi_uldivmod			8	<pre>char writeBuf[64] = {"Microchip 24CW160 EEPROM Driver"}</pre>			
aeabi_unwind_cpp_pr0	11 // page size is 32 bytes for 24CW160 EEPROM			shar material = [nicrosnip zremios contor briver]			
aeabi_unwind_cpp_pr1	12 // page count is 64 pages for 24CW160 EEPROM		9	char readbui[04];			
aeabi_unwind_cpp_pr2	13 CONST UNSIGNED INT PAGE_SIZE = 32; 14 const unsigned int PAGE_NUM = 64;		10				
_ascii_mbtowc	15		11	// page size is 32 bytes for 24CW160 FEPROM			
ascii_wctomb	16 unsigned int startPage;		10	// page size is sz bytes for 240000 EERON			
cxa_allocate_exception	17 unsigned int endPage;		12	// page count is 64 pages for 24CW160 EEPROM			
cxa_begin_catch	18 unsigned int offset;		13	const unsigned int PAGE SIZE = 32;			
cxa_begin_cleanup	20 unsigned int paddroosition:		14	const unsigned int PAGE NUM = 64			
cxa_call_terminate	21 unsigned int MemAddress;		17	const unsigned int the _ tot,			
cxa_call_unexpected	22 unsigned int bytesremaining;		15				
cxa_end_catch	23 unsigned int data_indx;		16	unsigned int startPage;			
cxa_end_cleanup	24 = 25 \pm void setun() {		17	unsigned int endPage			
cxa_free_exception	- 26 // RX0/TX0		1/	unsigned inc end age,			
cxa_get_globals	■ 27		18	unsigned int offset;			
cxa_get_globals_tast	28 // Wire1 = SDA1 & SCL1		19	unsigned int numberofpages:			
cxa_guard_abort	29 ± Wirel.begin();		20	unsigned int paddrossition;			
cxa_guard_acquire	■ 31		20	unsigned interpadd position,			
cxa_guaru_release	— ● 32		21	unsigned int MemAddress;			
cxa_init_primary_exception	■ 33 EEPROM_Read(0,0,readBuf,strlen(writeBuf));		22	unsigned int bytesremaining;			
cxa_pure_virtual()	34 ⊞ Seriali.printin(readBut);		23	unsigned int data indy.			
cxa_ternow	36		25	unsigned int data_indx,			
cxa_tune_match			24				
	▶ • 38 ± }		• 25	<pre> + void setup() { </pre>			
nsole		× Memory 1 @ 00000000	26				
le.Open: completed in 103 ms		▲ 00000000					
ogram segments:		00000010	• 27	Endii.Degin(115200);			
Auuress Size Code	nu vara KW Vala ZI Vala Pig	00000020		// Wirel = SDA1 & SCL1			
805CE64 0 0	0 0 0 R	00000040		+ Wirel hegin():			
8040000 118 372 117 480	892 0 0 R E	00000050					
4000000 4 712 0	0 4 712 0 RW	00000060	30	<pre> + W1re1.setClock(400000); </pre>			
	⊍ 2/8 528	00000070		EEPROM Write(0.0.writeBuf.strlen(writeBuf)):			
001268 24 0	0 24 0 RW	00000000		= do] ov(5) .			
	0 0 47 264 RW	000000A0	- JZ				
4001280 47 264 0		000000B0	0 33	<pre> EEPROM_Read(0,0,readBuf,strlen(writeBuf)); </pre>			
4001280 47 264 0	892 347 776 47 264	000000C0					
4001280 47 264 0 Total: 513 412 117 480		00000000	9 34 1	+ Seriali.println(readBut):			
4001280 47 264 0 Total: 513 412 117 480 r further information on FLF file data	a sections, execute command Elf.PrintSectionInfo(θ).	0000000	= 34 [E Seriali.printin(readBut);			
74001280 47 264 0 Total: 513 412 117 480 For further information on ELF file data bug.ReadIntoInstCache: updated instruction	a sections, execute command Elf.PrintSectionInfo(0). ction information within 1 code ranges (0x08040000-0x0805CAE8)	000000D0 000000E0 000000F0		<pre></pre>			
40001280 47 264 0 Total: 513 412 117 480 rr further information on ELF file data. bug.ReadIntoInstCache: updated instru nd.SourceFile ("Disabled output of control ou	a sections, execute command Elf.PrintSectionInfo(0). ction information within 1 code ranges (0x08040000-0x0805CAE8) ntrol characters");	000000000 0000000E0 0000000F0 000000100	• 34 [• 35 [· 36	<pre> E Seriall.print(readBut); E } </pre>			
24001280 47 264 0 Total: 513 412 117 480 .r further information on ELF file dat: bug.ReadIntoInstCache: updated instruction. ud.SourceFile ("Disabled output of con le.Open ("/home/fred/Arduino/24CW160_:	a sections, execute command Elf.PrintSectionInfo(0). ction information within 1 code ranges (0x08040000-0x0805CAE8) ntrol characters"); i2c/24CW160_i2c.ino");	00000000	• 34 (• 35 (• 36	<pre> H Seriall.printin(readBut); H yoid loop() { </pre>			



CPU halted.



Debug Views

Digikey

	Ozone - The J-Link Debugger V3.34 - /tmp/arduino/sketche	s/530D04C0491F47E86D0AE93F1255957D/24CW160_i2c.ino.elf	File	View Fin	d Debua	Tools W	/indow Hel	lp
<u>File View Find Debug Tools Window</u>	Help			Coporal				г
🛚 😃 🕨 Ke 🕶 🚓 ‡ 🏦			U 🗸	General				
Functions	24CW160 i2c.ino × V main.cpp × V		Eunctio	💽 Break &	Tracepoints	Alt+Shif	t+B 🏲	
Name		f actum	Functio		1 - C		[]	ma
(anonymous namespace)::pool::allocate		J setup	– Name	🗐 🗐 Call Sta	c <u>k</u>	Alt+Shif	t+K ⊨	
(anonymous namespace)::pool::free	25 ± void setup() { 26 // RX0/TX0					A 11 - OL 17		1
Unwind_Backtrace	● 27 王 Serial1.begin(115200);		(ano	Disasse	mbly	Alt+Shit	t+D 🗎	-
Unwind_ForcedUnwind	28 // Wirel = SDA1 & SCL1		(ano	Global I	Data	Alt+Chif	440	1
Onwind_Resume_or_Retnrow	29 ± Wirel.begin(); 30 ± Wirel.setClock(400000):		(Jala	AILTOIN		2
aeabi_div0	■ 31 EEPROM_Write(0,0,writeBuf,strlen(writeBuf));			Local D	ata	Alt+Shif	t+I	3
aeabi Idiymod	● 32 delay(5);		ι		ana	740.0111		4
aeabi uldivmod	■ 33 ± EERKOM_Read(0,0,readBuf,strten(writeBuf)); ■ 34 ± Serial1 printlp(readBuf).			Memor	1			17
aeabi_unwind_cpp_pr0	● 35 ± }							5
aeabi_unwind_cpp_pr1	36		ae	💌 <u>R</u> egiste	rs		► ►	6
aeabi_unwind_cpp_pr2	37 Vota toop() {							7
_ascii_mbtowc	39		ae	<u>⇒ T</u> ermina	d d	Alt+Shif	t+T	8
ascii_wctomb	40 // function to calculate the number of remaining bytes to read/write		ae	Weteba	d Doto			ŏ
cxa_allocate_exception	41 Unsigned int bytestert(unsigned int size, unsigned int offset) 42 {			w watche	u Dala		· · · ·	9
cxa_begin_cleanup	● 43 if((size + offset) <page_size)< td=""><td></td><td></td><td>Advanced</td><td></td><td></td><td></td><td>1</td></page_size)<>			Advanced				1
cxa call terminate			ae					1
cxa call unexpected	46 }			Code P	rofile	Alt+Shif	t+P	13
cxa_end_catch	47 else		de	Dette C	and the second sec	Alter Child		1
cxa_end_cleanup			ae	Data Sa	m <u>p</u> iing	Alt+Shir	(+Y	1
cxa_free_exception	50 }				on Traco	Alt⊥Chif	141	14
cxa_get_globals	51 }		a		UII Hace	Altronii		1
cxa_get_globals_tast	52 53 word EEDDOM Write/uncigned int page uncigned int offset shar #data uncigned	int cita)	as	Power 3	Sampling	Alt+Shif	t+A	10
CXa_guard_acquire	55 Void EErkom_write(disigned int page, disigned int offset, char "data, disigned i 54	Int Size)			<u>an ping</u>	7 11 - 01 11		1
cxa_guard_release	55 // calculate the beginning bit of the page addressing bits (paddrposition)			📃 Timelin	9	Alt+Shif	t+N	1.
cxa init primary exception	56 // p = page addressing bits		CX	Charlie -				10
cxa_pure_virtual()	58 // A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0		CX	Static				1
cxa_rethrow	59 // pppppssss			🗔 Call Gr	nh	Alt+Shif	t+H	2
cxa_throw	<pre>0 0 ± paddrposition = 0x05; 61</pre>		CX		φ <u>u</u>	740.0111		2
(4)	62 // calculate the start page and the end page		CX	Find Re	sults	Alt+Shif	t+E	2
Console	×	Memory 1 @ 00000000						4
AP[0]: Skipped ROMBASE read. CoreBaseAddr	manually set by user	000000000 E 7B CO 9B 8E 28 37 FD 3A 0E A8 C4 D9 89 F9 00 AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA	7ýCX	fx Eunctio	ns	Alt+Shif	t+F	Ζ.
AP[0]: Core found	do: 0x41 (ADM)	00000010 F7 8B CC 8D CD 2B B9 AD 24 0C 20 11 8D 11 C0 00 ÷.1.1+	: cx	- Maman	lleese	Alter Child		24
Cache: L1 I/D-cache present	UC. VAT (MUI)	00000030 2A 33 B0 AD B6 65 68 02 41 06 32 E0 94 0E 08 10 *3°.¶e	h	Memor	0 <u>s</u> age	Ait+Shit	175	2
Found Cortex-M7 r1p1, Little endian.		00000040 FC 37 DB BE 6F 74 5A 1C 06 00 B5 46 81 65 45 54 ü7Ü3ot	zCX	Source	Filos	∆lt+Shif	t+11	21
FPUnit: 8 code (BP) slots and 0 literal s	ots lly set by user: AvEGGEEGGG	00000050 EA C4 DB 5F 37 9D 98 C5 25 0C D8 93 83 99 42 E0 ÊAU 7.	° cx	C Source	riica	Altronii		2
I-Cache L1: 16 KB, 256 Sets, 32 Bytes/Lin	e, 2-Way	000000070 6F FD DB 6F 3F D6 C7 BA 88 02 0C 50 80 A5 3C 1A ov00?	C°	Informatio				4
D-Cache L1: 16 KB, 128 Sets, 32 Bytes/Lin	, 4-Way	00000080 FD FC FA FB FD 77 D1 43 9A 8A C8 4A 26 96 0D 40 ýײַַּטְׁטְׁאַ	Ńc	Connel		Alter Child		2
Connected to target device.		00000090 2C D6 DA 52 FE EF E7 FF 09 C0 70 56 62 11 14 40 ,0URbi	çý CX	<u>Consol</u>	;	Alt+Shir	(+C	- 21
Reset: Halt core after reset via DEMCR.VC	CORERESET.	0000000B0 F9 7D 83 EF 7F F8 67 B9 42 08 4B 24 90 01 C5 0E ù}.i.a	g ¹ ev	No. 1. Link (ontrol Danol	Alt+Shif	t+1	3
Reset: Reset device via AIRCR.SYSRESETREQ		000000C0 4A FF 9E 5D BA CC 8F CF 17 04 88 01 D1 7B 29 3C Jy.]	.ï^		onuorranei	Altronii		3
J-LINK: Flash download: Bank 0 @ 0x080000 Memory map 'after startup completion poin	u: skipped. contents already match	00000000 6F 94 D5 E6 DC E1 BE 7D 49 01 02 F0 20 80 90 29 0.0æU	Δ _CX					5
Startup complete (PC=0x080427FC)	=	000000F0 FE DD EC B8 E7 77 A0 DE 96 58 C2 20 80 12 44 48 bY1 cv		Toolbar	S		•	3.
Debug.Continue();		00000100 7E 25 DB 5D 0B 1B F9 56 F2 A1 41 5B 13 3C 68 01 ~%Û].	ùV	Enter F	III Coroon	AltaChie	t+ Doturn	3
window.snow ("Global Data");		100000110 FB F6 FC DB D2 95 FF C3 34 1C 8A A2 96 02 20 02 ນິວິນັບປ. 00000120 FD FF CD F8 85 C7 FA F3 4R RF 60 90 64 40 92 82 ນີ່ງັ້ດ (YA CX	EnterF	un screen	AILTSHI	t+Return	3
•		00000130 C1 99 55 9D DF EE 5C C6 64 08 83 54 16 81 BD 5D Å.U.B	\Æ	a rothrow				21
CPU halted.					Ln 34 Ch 28	Connected @ 4	MHz	



Show Global Data



	Ozone - The J-Link Debugger V3.34 - /tmp/arduino/sketches	1F47E86D0AE93F1255957D/24CW160_i2c.ino.elf 0 ×							
File View Find Debug Tools Window	Help	Glob	Giobal Data						
. U		Nam	ne 😽	Value	Location	Size	Туре	Access	Scope 🔺
Functions	× 24CW160_i2c.ino × main.cpp ×	= re	eadBuf	"Microchip 240	2400 12B4	64	char[64]		24CW160 i2c.ino.cpp
Name	▲ File Scope ♦	f setup	[0]	77 ('M')	2400 1284	1	char		24CW160_i2c ino cpp
(anonymous namespace)::pool::allocate	● 25		[9]	105 (141)	2400 1204		char		24CW/160_i2c inc cpp
Unwind Backtrace	26 // RX0/TX0		[1]	105 (11)	2400 1285	1	char		24CW100_12C.in0.cpp
Unwind ForcedUnwind	28 // Wirel = SDA1 & SCL1		[2]	99 ('c')	2400 1286	1	char		24CW160_12c.ino.cpp
Unwind_Resume_or_Rethrow	29 ⊞ Wirel.begin();		[3]	114 ('r')	2400 12B7	1	char		24CW160_i2c.ino.cpp
aeabi_atexit	<pre>31</pre>		[4]	111 ('o')	2400 12B8	1	char		24CW160_i2c.ino.cpp
aeabi Idivmod	■ 32 → delay(5);		[5]	99 ('c')	2400 12B9	1	char		24CW160 i2c.ino.cpp
aeabi_uldivmod	<pre>33 EPrior Read(0,0,1eadBd),strten(writeBd)); 34 F Seriall.println(readBuf);</pre>		[6]	104 ('h')	2400 1284	1	char		24CW160_i2c ino cpp
aeabi_unwind_cpp_pr0	● 35 ± }		[9]	105 (131)	2400 1200		char		24CW/160_i2c inc opp
aeabi_unwind_cpp_pr1	37 void loop() {		1/1	105 (.1.)	2400 1288	1	char		240W100_120.in0.cpp
ascii mbtowc	● 38 ± }		[8]	112 ('p')	2400 12BC	1	char		24CW160_12c.ino.cpp
ascii_wctomb	40 // function to calculate the number of remaining bytes to read/write		[9]	32 (' ')	2400 12BD	1	char		24CW160_i2c.ino.cpp
cxa_allocate_exception	41 unsigned int bytesleft(unsigned int size, unsigned int offset)		[10]	50 ('2')	2400 12BE	1	char		24CW160_i2c.ino.cpp
cxa_begin_catch	42 1 43 ⊞ if((size + offset) <page size)<="" td=""><td></td><td>[11]</td><td>52 ('4')</td><td>2400 12BF</td><td>1</td><td>char</td><td></td><td>24CW160 i2c.ino.cpp</td></page>		[11]	52 ('4')	2400 12BF	1	char		24CW160 i2c.ino.cpp
cxa_call_terminate	44 {		[12]	67 ('C')	2400 1200	1	char		24CW160_i2c ino cpp
cxa_call_unexpected	45 H return size;		[12]	07 (C)	2400 1200		char		24CW/160_i2e ine opp
cxa_end_catch	47 else		[13]	87 ('W')	2400 1201	1	char		24CW100_12C.IIIO.cpp
cxa_end_cleanup	48 1 ● 49 ⊞ return PAGE SIZE - offset;		[14]	49 ('1')	2400 12C2	1	char		24CW160_i2c.ino.cpp
cxa_ree_exception	50 }		[15]	54 ('6')	2400 12C3	1	char		24CW160_i2c.ino.cpp
cxa_get_globals_fast	51 }		[16]	48 ('0')	2400 12C4	1	char		24CW160_i2c.ino.cpp =
cxa_guard_abort	53 void EEPROM_Write(unsigned int page, unsigned int offset, char *data, unsigned in	nt size)	[17]	32 (' ')	2400 1205	1	char		24CW160 i2c.ino.cpp
cxa_guard_acquire	54 ± 1 55 // calculate the beginning bit of the page addressing bits (paddrposition)		[18]	69 ('E')	2400 1206	1	char		24CW160_i2c ino con
cxa_guard_release	56 // p = page addressing bits		[10]		2400 1200		char		24CW/160_i2e ine opp
ora	5/ // s = page size bits 58 // A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0		[19]	69 ('E')	2400 1207	1	char		24CW160_12C.110.cpp
cxa_rethrow	59 // pppppssss		[20]	80 ('P')	2400 12C8	1	char		24CW160_i2c.ino.cpp
cxa_throw	 60 ± paddrposition = 0x05; 61 		[21]	82 ('R')	2400 12C9	1	char		24CW160_i2c.ino.cpp
	62 // calculate the start page and the end page		[22]	79 ('0')	2400 12CA	1	char		24CW160_i2c.ino.cpp
Console	x	Memory 1	[23]	77 ('M')	2400 12CB	1	char		24CW160 i2c.ino.cpp
AP[0]: Skipped ROMBASE read. CoreBaseAdd AP[0]: Core found	<pre>manually set by user</pre>	00000000	[24]	32 (' ')	2400 1200	1	char		24CW160_i2c.ino.con
CPUID register: 0x411FC271. Implementer	code: 0x41 (ARM)	00000020	[25]	52 () 60 (IDI)	2400 1200	1	char		24CW160_i2c inc cpp
Cache: L1 I/D-cache present		00000030	[25]	08 ('U')	2400 1200	1	char		24CW100_12C.ino.cpp
FPUnit: 8 code (BP) slots and 0 literal	slots	00000050	[26]	114 ('r')	2400 12CE	1	char		24CW160_12c.ino.cpp
ROM table scan skipped. CoreBaseAddr man	Jally set by user: 0xE00FE000	00000060	[27]	105 ('i')	2400 12CF	1	char		24CW160_i2c.ino.cpp
D-Cache L1: 16 KB, 128 Sets, 32 Bytes/Li	ie, 4-Way	00000080	[28]	118 ('v')	2400 12D0	1	char		24CW160_i2c.ino.cpp
Connected to target device.		00000090	[29]	101 ('e')	2400 12D1	1	char		24CW160 i2c.ino.cpp
Reset: Halt core after reset via DEMCR.V	CORERESET.	000000B0	[30]	114 ('r')	2400 1202	1	char		24CW160_i2c.ino.con
Reset: Reset device via AIRCR.SYSRESETRE). MA: Skinned Contents already match	00000000	[20]		2400 1202		char		24CW160_i2c inc.cpp
Memory map 'after startup completion poi	t' is active	000000E0	[21]	0 (\0)	2400 1203	1	char		240W100_120.in0.cpp
Startup complete (PC=0x080427FC) Debug Continue():		000000F0	[32]	0 ('\0')	2400 12D4	1	char		24CW160_12c.ino.cpp
Window.Show ("Global Data");		00000110	[33]	0 ('\0')	2400 12D5	1	char		24CW160_i2c.ino.cpp
•	· · · · · · · · · · · · · · · · · · ·	00000120							Image: A start of the start

Ln 34 Ch 28 Connected @ 4 MHz

23



Show Watch Data

		Ozone - T	he J-Link Debugger V3.34 - /tmp/ar	duino/sketches/53	30D04C0491F4	7E86D0AE93F12	255957D/24CW160_	i2c.ino.elf				_ 0 X
File View Find Debug Tools Window	Help											
Functions	× Carolina									Disassembly		
Name	A / 24CW1	60_12c.ino × v main.cpp × \								Wirel.beg	inTransmission(EEPROM ADDR):
(anonymous namesnace)::nool::allocate	😑 [📄 File S			‡ [f	EEPROM_Write					08040392	MOVS	R1, #80
(anonymous namespace)::pool::free	53	void EEPROM_Write(unsigned int	page, unsigned int offset, char *d	lata, unsigned int	size)				[MemAddres	s = startPage <	< paddrposition
Unwind Backtrace	• 54 ±	• {								08040394	LDR	R3, [R5] R2 [R2]
Unwind ForcedUnwind	55	<pre>// calculate the beginning b // n = page addressing bits</pre>	oit of the page addressing bits (pad	arposition)						Wirel.beg	inTransmission(EEPROM ADDR);
Unwind Resume or Rethrow	57	<pre>// s = page size bits</pre>								• 08040398	LDR	R0, =Wirel
aeabi atexit	58	// A10 A9 A8 A7 A6 A5 A4 A3	A2 A1 A0							MemAddres	s = startPage <	< paddrposition
aeabi_idiv0	59	// p p p p p s s	S S S							08040394	ORR.W	R3, R2
aeabi_ldivmod	61	paddrposition = 0x03;		Watche	d Data 1			X		080403A0	STR.W	R3, [R11]
aeabi_uldivmod	62	<pre>// calculate the start page</pre>	and the end page	-						if((size	+ offset) <page< td=""><td>STZE)</td></page<>	STZE)
aeabi_unwind_cpp_pr0	• 63 ±	<pre>startPage = page;</pre>		Express	sion	Value		Loca		Destinters 1/C	DUD	
aeabi_unwind_cpp_pr1	• 64 ±	endPage = page +((size + off	<pre>set)/PAGE_SIZE);</pre>	at a st D a		0		2400.21		Registers 1 (C	PU)	lua Description
aeabi_unwind_cpp_pr2	66	// number of pages to be wri	itten	stanpa	je 🛛	0		2400 1.		Name	Va 607 Degisters	lue Description
ascii_mbtowc	● 67 ±	numberofpages = (endPage-sta	artPage) + 1;	endPag	e	0		2466 1	-	H CPU	007 Registers	CPU Registers
ascii_wctomb	68	// set writeBuf array index	to 0x00	chur ag	<u> </u>	v		2400 1		=		
cxa_allocate_exception	09 ±	data_indx = 0x00;		offset		0						
CXa_begin_cleanun	71	// write the data to EEPROM		number	ofnogoo			2400 1				
cxa_begin_cleanup	• 72 ±	<pre>for(int i=0; i<numberofpages< pre=""></numberofpages<></pre>	;; i++)	number	oipages	1		2400 1.		=		
cxa_call_unexpected	73	{	the byte address to	paddrpo	osition	5		2400 1				
cxa end catch	75	// calculalte the beginnin	ng memory location	padaip		-		2400 1				
cxa end cleanup	● 76 ±	<pre>MemAddress = startPage <<</pre>	paddrposition offset;	MemAd	dress	0		2400 1				
cxa_free_exception	77	<pre>// calculate remaining num</pre>	ber of bytes to be written	butoero	maining	0		2400 1				
cxa_get_globals	79	bytesremaining = bytestert	(Size,offset);	bytesre	maining	U		2400 1.				
cxa_get_globals_fast		Wirel.beginTransmission(EE	EPROM ADDR);	data in	dx	0		2400 1				
cxa_guard_abort	• 81 ±	Wirel.write(highByte(MemAd	dress));			•		2400 1.				
cxa_guard_acquire	• 82 ±	Wirel.write(lowByte(MemAdd	iress));									
cxa_guard_release	 84 + 	for(int i=0: i <bvtesremain< td=""><td>ning: i++)</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></bvtesremain<>	ning: i++)									
cxa_init_primary_exception	85	{	57.5									
cxa_pure_virtual()	86 +	Wirel.write(writeBuf[dat	ta_indx++]);									
cxa_rethrow	87	} Wirel endTransmission():										
CXa_trilow	▼ 89											
) • 90 🛨	<pre>startPage += 1; // increme</pre>	ent the page								111	
Console				× Me	emory 1 @ 00000	000						3
AP[3]: AHB-AP (IDR: Not set)				▲ 00	000000 DE 7B	C0 9B 8E 28 37	FD 3A 0E A8 C4 D9	89 F9 00 [A(7ý:.¨ÄÙ.ù.			
AP[0]: Skipped ROMBASE read. CoreBaseAdd	r manually	set by user		00	000010 F7 8B	CC 8D CD 2B B9	AD 24 0C 20 11 8D	11 CO 00 ÷.I.I+	-1.\$A.			
CPUID register: 0x411FC271. Implementer	code: 0x41	(ARM)		00	000020 IF 55	B0 AD B6 65 68	02 41 06 32 E0 94	0E 08 10 *3°. ¶e	h.A.2à			
Cache: LI I/D-cache present				00	000040 FC 37	DB BE 6F 74 5A	1C 06 00 B5 46 81	65 45 54 ü7Û¾ot	ΖμF.eET			
Found Cortex-M7 r1p1, Little endian.	alata			00	000050 EA C4	DB 5F 37 9D 98	C5 25 0C D8 93 83	99 42 E0 êÂŬ_7.	.Å%.ØBà			
ROM table scan skipped CoreBaseAddr man	ually set b	V USEC: 0xE00EE000		00	0000000 DF /8	JF /A DB BB DF	5F 5A 9A 86 2C 09	00 A9 40 BX/ZU»	15_2,¢@			
I-Cache L1: 16 KB, 256 Sets, 32 Bytes/Lin	ne, 2-Way	,		00	000080 FD FC	FA FB FD 77 D1	43 9A 8A C8 4A 26	96 0D 40 ýüúûýw	ŃCÈJ&@			
D-Cache L1: 16 KB, 128 Sets, 32 Bytes/Li	ne, 4-Way			00	000090 2C D6	DA 52 FE EF E7	FF 09 C0 70 56 62	11 14 40 ,ÖÚRÞI	çÿ.ÀpVb@			
connected to target device.				00	0000A0 E9 9D	5F 83 9F 79 D7	UF 90 F8 01 35 2A	1A 9E 44 ey	/×15.ø.5*D			
Reset: Halt core after reset via DEMCR.V	C CORERESET	·.		00	0000C0 4A FF	9E 5D BA CC 8F	CF 17 04 88 01 D1	7B 29 3C Jÿ.1ºÌ				
Reset: Reset device via AIRCR.SYSRESETRE	Q.			00	0000D0 6F 94	D5 E6 DC E1 BE	7D 49 01 02 F0 20	80 90 29 o.ÕæÜá	l₄}Ið)			
J-Link: Flash download: Bank 0 @ 0x080000	000: Skippe	ed. Contents already match		00	0000E0 74 DC	FD 75 C9 FD C5	21 A0 01 4C D9 19	8B 83 03 tÜýuÉý	A!LÚ			
Startup complete (PC=0x080427FC)	inc is acti	ve		00	0000F0 FE DD	EC 88 E/ // A0 DB 5D 0B 1B F9	56 F2 A1 A1 5R 13	12 44 48 PY1 CW	ùνà:Δ[. <h.< td=""><td></td><td></td><td></td></h.<>			
Debug.Continue();				00	000110 FB F6	FC DB D2 95 FF	C3 34 1C 8A A2 96	02 20 02 ûöüÛÒ.	ÿÃ4¢			
x					000120 FD EF	CD F8 85 C7 EA	F3 4B BE 60 90 64	40 92 82 ýïÍø.Ç	êóK¾`.d@			
F				00	000130 C1 99	55 9D DF EE 5C	Lb 64 08 83 54 16	81 BD 50 A.U.G1	\#01*I			

Ln 72 Ch 37 Connected @ 4 MHz



Show Variable Data - Array



	8		char writ	- n	fleat fund-secolds	04CH3C0 5500	OM Driver"}:
	9		char read	۲	Set Breakpoint	F9	,,,
	10			۲	Break on Change		
	11		// page s		Set Tracepoint (Start)		
	13		const uns		Set Tracepoint (Stop)		
	14		const uns	-	Set Hacepoint (Stop)		
	15			<u> </u>	Show Definition	F12	
	16		unsigned		Show Declaration	Shift+E12	
	1/		unsigned		Chew Deta	01117	
	19		unsigned	10	Show Da <u>t</u> a	Ctrl+1	
	20		unsigned	20	Show Call Graph	Ctrl+H	
	21		unsigned	B	Show in Memory Map	Ctrl+B	
	22		unsigned	_			
	23		unsigned	Q	<u>W</u> atch	Ctrl+W	
	24	+	void setu	9	Quick Watch	Shift+F9	
	26		// RX0/				
•	27	+	Serial1	-	Go To <u>P</u> C	Ctrl+P	
	28		// Wire		Go To <u>L</u> ine	Ctrl+L	
	29	H	Wirel.b		Fi <u>n</u> d	Ctrl+F	
	30	+ +	EEPROM		Find In Trace	Ctrl+Shift+T	
	32	Ŧ	delay(5				1
•	33	+	EEPROM	Ē	E <u>x</u> pand All	Alt++	
•	34	+	Serial1		Cut	Ctrl+X	
•	35	+	}		Cui	CUITA	
	30		void loon	41	<u>C</u> opy	Ctrl+C	
	38	+	}		Paste	Ctrl+V	
			-		Line Numbers		
					Execution Countors	Ctrite	
					Execution Counters	CUITE	
in	duir	10/	hardware/m		Instruction Encodings		initions.h");
а	rdui	no,	/hardware/	~	Pseudo Instructions		finitions.h")
(n)	duir	10/	hardware/m	*	Export		rduino.h");

Memory 1 @	<u>)</u> 23	FFF	FD4	Ļ													
23FFFFD4																	
23FFFFE4																	
23FFFFF4													00	00	00	00	
24000004	4D	69	63	72	6F	63	68	69	70	20	32	34	43	57	31	36	Microchip 24CW16
24000014	30	20	45	45	50	52	4F	4D	20	44	72	69	76	65	72	00	0 EEPROM Driver.
24000024	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000034	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000044	17	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000054	09	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000064	03	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000074	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000084	98	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000094	07	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240000A4	ЗD	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	=
240000B4	14	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240000C4	18	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240000D4	19	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240000E4	A1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	j
240000F4	9A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24000104	9B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	



Arduino Pro Primer Debugging Arduino Sketches Debug the Microchip 24CW160 EEPROM Driver

Show Variable Data - Array

8	char writ	.n	fleat (mainschie	346H360 5500	<pre>OM Driver"};</pre>
9	char read	۲	Set <u>B</u> reakpoint	F9	
10			Break on <u>C</u> hange		
11	// page s		Set Tracepoint (Start)		
12	const uns	-	Set Tracopoint (Stop)		
14	const uns		Set Hacepoint (Stop)		
15		8	Show Definition	F12	
16	unsigned	•=	Show Declaration	Shift+E12	
1/	unsigned	E Rh	Cherry Deta	OHINT IZ	
19	unsigned	10	Snow Da <u>t</u> a	Ctrl+1	
20	unsigned	чĩ	Show Call Grap <u>h</u>	Ctrl+H	
21	unsigned	ß	Show in <u>M</u> emory Map	Ctrl+B	
22	unsigned	@	Watch	Ctrl+W	
24		٩	Ouick Watch	Shift+E9	
• 25 ±	void setu	Ÿ	Quick Watering	0111111	
≥0 ≥ 27 ∓	Serial1	⇒	Go To <u>P</u> C	Ctrl+P	
28	// Wire		Go To Line	Ctrl+L	
29 +	Wirel.b		Find	Ctrl+F	
30 ±	Wirel.s		Find In Trace	Ctrl+Shift+T	
• <u>31</u> +	del av (5		rin <u>a</u> in riace	Carronner	;
• 33 F	EEPROM	¢	E <u>x</u> pand All	Alt++	
34 +	Serial1				
 35 + 	}	_	Cut	Ctrl+X	
36	unid loop	Û	<u>C</u> opy	Ctrl+C	
● 38 ±	}		Paste	Ctrl+V	
			Line Numbers	•	
			Execution Counters	Ctrl+E	
and solve a st	h		Instruction Encodings		Laisiana huy
arduino/	hardware/n	4	Pseudo Instructions		finitions.h");
aruutilu	/ nar uwar e/	_			11111110113.11)
rduino/	hardware/n	*	Export		rduino.h");

Memory 1 @	24	0012	2B4														
240012B4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240012C4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240012D4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240012E4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240012F4	00	00	00	00	84	A1	05	08	00	00	00	00	E8	03	00	00	įè
24001304	00	00	00	00	00	00	00	00	00	00	00	00	1B	00	74	00	t.
24001314	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001324	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001334	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001344	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001354	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001364	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001374	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001384	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
24001394	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240013A4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240013B4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240013C4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240013D4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
240013E4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	



Find Arduino Source Files



27



Visual Studio Code Works Too





New Project Wizard	×
Program File Choose the Program to be debugged	
LF, Motorola S-record, Intel Hex, or Binary file (optional)	
/home/fred/24CW160/build/24CW160.ino.elf	

X Cancel

< <u>B</u>ack

28



Arduino Pro Primer

Next Time...

DigiKey

MORE TO COME..

Thank you for attending!!!

Please consider the resources below:

- Today's Download Package
- arduino.cc







Thank You





SALANA.

