

Custom AVR Assembler

Caleb Jurgensen

Objectives

- Make an AVR assembler
- Program an Arduino Uno with assembly

AVR

- A family of 8-bit RISC microcontrollers developed by Atmel
 - Use the AVR architecture and instruction set
- Arduinos generally use these
 - Arduino Uno uses the atmega328p
- AVR32 is a 32 bit version

Turning on an LED

- Pins are controlled by registers
- Uno's LED is on PB5
- Assembly

11.4.2 PORTB – The Port B Data Register

Bit	7	6	5	4	3	2	1	0	PORTB
Output (0x25)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	
Read/Write	RW	RW	RW	RW	RW	RW	RW	RW	
Initial Value	0	0	0	0	0	0	0	0	

11.4.3 DDRB – The Port B Data Direction Register

Bit	7	6	5	4	3	2	1	0	DDRB
Output (0x24)	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0	
Read/Write	RW	RW	RW	RW	RW	RW	RW	RW	
Initial Value	0	0	0	0	0	0	0	0	

```

#include "./m328Pdef.inc"

ldi r16,0b00100000
out DDRB,r16 ; Set data direction to out
out PortB,r16 ; Set pins high or low
Start:
rjmp Start ; Loop

```

Making it blink

- Need a counter that counts to roughly 2 million for 1Hz blinking
- Spread the number across three registers
- add and adc set a carry flag. adc also adds this bit

```
Increment:
add numLow, one ; increment lower by one
adc numMid, zero ; add zero + carry bit to mid
adc numUpper, zero ; same for upper
cpi numUpper, 0b00011111 ; half of full register
breq TurnOn ; branch if numUpper is Same or Higher than v
alu in previous instruction
rjmp Increment ; else increment again
```

```
.include "../m328pdef.inc"
.equ ledMask = 0b00100000
.def zero = r1 ; r1 should always be zero
.def numLow = r16
.def numMid = r17
.def numUpper = r18
.def one = r19
.def temp = r20

ldi one, 1 ; initialize our "one" register
ldi temp, 0 ; make sure zero register is 0
mov zero, temp
ldi temp, ledMask
out DDRB, temp ; Set led pin data direction to out
out PortB, temp ; Set led pin to high

Increment:
add numLow, one ; increment lower by one
adc numMid, zero ; add zero + carry bit to mid
adc numUpper, zero ; same for upper
cpi numUpper, 0b00011111 ; half of full register
breq TurnOn ; branch if numUpper is Same or Higher than value in previous instruction
rjmp Increment ; else increment again

TurnOn:
ldi temp, ledMask
out PortB, temp ; Set pin to high
clr numLow
clr numMid
clr numUpper

Increment2:
add numLow, one ; increment lower by one
adc numMid, zero ; add zero + carry bit to mid
adc numUpper, zero ; same for upper
cpi numUpper, 0b00011111 ; half of full register
breq TurnOff ; branch if numUpper is Same or Higher than value in previous instruction
rjmp Increment2 ; else increment again

TurnOff:
mov temp, zero
out PortB, temp ; Set pin to low
clr numLow
clr numMid
clr numUpper
rjmp Increment ; start incrementing again
```

- Arduino compiles this program
- Into 385 assembly instructions

```
#include <Arduino.h>

int led = 13;
void setup() {
  pinMode(led, OUTPUT);
}
void loop() {
  digitalWrite(led, HIGH);
  delay(1000);
  digitalWrite(led, LOW);
  delay(1000);
}
```

```
1  .org 0x00000000
2  .org 0x00000001
3  .org 0x00000002
4  .org 0x00000003
5  .org 0x00000004
6  .org 0x00000005
7  .org 0x00000006
8  .org 0x00000007
9  .org 0x00000008
10 .org 0x00000009
11 .org 0x0000000A
12 .org 0x0000000B
13 .org 0x0000000C
14 .org 0x0000000D
15 .org 0x0000000E
16 .org 0x0000000F
17 .org 0x00000010
18 .org 0x00000011
19 .org 0x00000012
20 .org 0x00000013
21 .org 0x00000014
22 .org 0x00000015
23 .org 0x00000016
24 .org 0x00000017
25 .org 0x00000018
26 .org 0x00000019
27 .org 0x0000001A
28 .org 0x0000001B
29 .org 0x0000001C
30 .org 0x0000001D
31 .org 0x0000001E
32 .org 0x0000001F
33 .org 0x00000020
34 .org 0x00000021
35 .org 0x00000022
36 .org 0x00000023
37 .org 0x00000024
38 .org 0x00000025
39 .org 0x00000026
40 .org 0x00000027
41 .org 0x00000028
42 .org 0x00000029
43 .org 0x0000002A
44 .org 0x0000002B
45 .org 0x0000002C
46 .org 0x0000002D
47 .org 0x0000002E
48 .org 0x0000002F
49 .org 0x00000030
50 .org 0x00000031
51 .org 0x00000032
52 .org 0x00000033
53 .org 0x00000034
54 .org 0x00000035
55 .org 0x00000036
56 .org 0x00000037
57 .org 0x00000038
58 .org 0x00000039
59 .org 0x0000003A
60 .org 0x0000003B
61 .org 0x0000003C
62 .org 0x0000003D
63 .org 0x0000003E
64 .org 0x0000003F
65 .org 0x00000040
66 .org 0x00000041
67 .org 0x00000042
68 .org 0x00000043
69 .org 0x00000044
70 .org 0x00000045
71 .org 0x00000046
72 .org 0x00000047
73 .org 0x00000048
74 .org 0x00000049
75 .org 0x0000004A
76 .org 0x0000004B
77 .org 0x0000004C
78 .org 0x0000004D
79 .org 0x0000004E
80 .org 0x0000004F
81 .org 0x00000050
82 .org 0x00000051
83 .org 0x00000052
84 .org 0x00000053
85 .org 0x00000054
86 .org 0x00000055
87 .org 0x00000056
88 .org 0x00000057
89 .org 0x00000058
90 .org 0x00000059
91 .org 0x0000005A
92 .org 0x0000005B
93 .org 0x0000005C
94 .org 0x0000005D
95 .org 0x0000005E
96 .org 0x0000005F
97 .org 0x00000060
98 .org 0x00000061
99 .org 0x00000062
100 .org 0x00000063
101 .org 0x00000064
102 .org 0x00000065
103 .org 0x00000066
104 .org 0x00000067
105 .org 0x00000068
106 .org 0x00000069
107 .org 0x0000006A
108 .org 0x0000006B
109 .org 0x0000006C
110 .org 0x0000006D
111 .org 0x0000006E
112 .org 0x0000006F
113 .org 0x00000070
114 .org 0x00000071
115 .org 0x00000072
116 .org 0x00000073
117 .org 0x00000074
118 .org 0x00000075
119 .org 0x00000076
120 .org 0x00000077
121 .org 0x00000078
122 .org 0x00000079
123 .org 0x0000007A
124 .org 0x0000007B
125 .org 0x0000007C
126 .org 0x0000007D
127 .org 0x0000007E
128 .org 0x0000007F
129 .org 0x00000080
130 .org 0x00000081
131 .org 0x00000082
132 .org 0x00000083
133 .org 0x00000084
134 .org 0x00000085
135 .org 0x00000086
136 .org 0x00000087
137 .org 0x00000088
138 .org 0x00000089
139 .org 0x0000008A
140 .org 0x0000008B
141 .org 0x0000008C
142 .org 0x0000008D
143 .org 0x0000008E
144 .org 0x0000008F
145 .org 0x00000090
146 .org 0x00000091
147 .org 0x00000092
148 .org 0x00000093
149 .org 0x00000094
150 .org 0x00000095
151 .org 0x00000096
152 .org 0x00000097
153 .org 0x00000098
154 .org 0x00000099
155 .org 0x0000009A
156 .org 0x0000009B
157 .org 0x0000009C
158 .org 0x0000009D
159 .org 0x0000009E
160 .org 0x0000009F
161 .org 0x000000A0
162 .org 0x000000A1
163 .org 0x000000A2
164 .org 0x000000A3
165 .org 0x000000A4
166 .org 0x000000A5
167 .org 0x000000A6
168 .org 0x000000A7
169 .org 0x000000A8
170 .org 0x000000A9
171 .org 0x000000AA
172 .org 0x000000AB
173 .org 0x000000AC
174 .org 0x000000AD
175 .org 0x000000AE
176 .org 0x000000AF
177 .org 0x000000B0
178 .org 0x000000B1
179 .org 0x000000B2
180 .org 0x000000B3
181 .org 0x000000B4
182 .org 0x000000B5
183 .org 0x000000B6
184 .org 0x000000B7
185 .org 0x000000B8
186 .org 0x000000B9
187 .org 0x000000BA
188 .org 0x000000BB
189 .org 0x000000BC
190 .org 0x000000BD
191 .org 0x000000BE
192 .org 0x000000BF
193 .org 0x000000C0
194 .org 0x000000C1
195 .org 0x000000C2
196 .org 0x000000C3
197 .org 0x000000C4
198 .org 0x000000C5
199 .org 0x000000C6
200 .org 0x000000C7
201 .org 0x000000C8
202 .org 0x000000C9
203 .org 0x000000CA
204 .org 0x000000CB
205 .org 0x000000CC
206 .org 0x000000CD
207 .org 0x000000CE
208 .org 0x000000CF
209 .org 0x000000D0
210 .org 0x000000D1
211 .org 0x000000D2
212 .org 0x000000D3
213 .org 0x000000D4
214 .org 0x000000D5
215 .org 0x000000D6
216 .org 0x000000D7
217 .org 0x000000D8
218 .org 0x000000D9
219 .org 0x000000DA
220 .org 0x000000DB
221 .org 0x000000DC
222 .org 0x000000DD
223 .org 0x000000DE
224 .org 0x000000DF
225 .org 0x000000E0
226 .org 0x000000E1
227 .org 0x000000E2
228 .org 0x000000E3
229 .org 0x000000E4
230 .org 0x000000E5
231 .org 0x000000E6
232 .org 0x000000E7
233 .org 0x000000E8
234 .org 0x000000E9
235 .org 0x000000EA
236 .org 0x000000EB
237 .org 0x000000EC
238 .org 0x000000ED
239 .org 0x000000EE
240 .org 0x000000EF
241 .org 0x000000F0
242 .org 0x000000F1
243 .org 0x000000F2
244 .org 0x000000F3
245 .org 0x000000F4
246 .org 0x000000F5
247 .org 0x000000F6
248 .org 0x000000F7
249 .org 0x000000F8
250 .org 0x000000F9
251 .org 0x000000FA
252 .org 0x000000FB
253 .org 0x000000FC
254 .org 0x000000FD
255 .org 0x000000FE
256 .org 0x000000FF
257 .org 0x00000100
258 .org 0x00000101
259 .org 0x00000102
260 .org 0x00000103
261 .org 0x00000104
262 .org 0x00000105
263 .org 0x00000106
264 .org 0x00000107
265 .org 0x00000108
266 .org 0x00000109
267 .org 0x0000010A
268 .org 0x0000010B
269 .org 0x0000010C
270 .org 0x0000010D
271 .org 0x0000010E
272 .org 0x0000010F
273 .org 0x00000110
274 .org 0x00000111
275 .org 0x00000112
276 .org 0x00000113
277 .org 0x00000114
278 .org 0x00000115
279 .org 0x00000116
280 .org 0x00000117
281 .org 0x00000118
282 .org 0x00000119
283 .org 0x0000011A
284 .org 0x0000011B
285 .org 0x0000011C
286 .org 0x0000011D
287 .org 0x0000011E
288 .org 0x0000011F
289 .org 0x00000120
290 .org 0x00000121
291 .org 0x00000122
292 .org 0x00000123
293 .org 0x00000124
294 .org 0x00000125
295 .org 0x00000126
296 .org 0x00000127
297 .org 0x00000128
298 .org 0x00000129
299 .org 0x0000012A
300 .org 0x0000012B
301 .org 0x0000012C
302 .org 0x0000012D
303 .org 0x0000012E
304 .org 0x0000012F
305 .org 0x00000130
306 .org 0x00000131
307 .org 0x00000132
308 .org 0x00000133
309 .org 0x00000134
310 .org 0x00000135
311 .org 0x00000136
312 .org 0x00000137
313 .org 0x00000138
314 .org 0x00000139
315 .org 0x0000013A
316 .org 0x0000013B
317 .org 0x0000013C
318 .org 0x0000013D
319 .org 0x0000013E
320 .org 0x0000013F
321 .org 0x00000140
322 .org 0x00000141
323 .org 0x00000142
324 .org 0x00000143
325 .org 0x00000144
326 .org 0x00000145
327 .org 0x00000146
328 .org 0x00000147
329 .org 0x00000148
330 .org 0x00000149
331 .org 0x0000014A
332 .org 0x0000014B
333 .org 0x0000014C
334 .org 0x0000014D
335 .org 0x0000014E
336 .org 0x0000014F
337 .org 0x00000150
338 .org 0x00000151
339 .org 0x00000152
340 .org 0x00000153
341 .org 0x00000154
342 .org 0x00000155
343 .org 0x00000156
344 .org 0x00000157
345 .org 0x00000158
346 .org 0x00000159
347 .org 0x0000015A
348 .org 0x0000015B
349 .org 0x0000015C
350 .org 0x0000015D
351 .org 0x0000015E
352 .org 0x0000015F
353 .org 0x00000160
354 .org 0x00000161
355 .org 0x00000162
356 .org 0x00000163
357 .org 0x00000164
358 .org 0x00000165
359 .org 0x00000166
360 .org 0x00000167
361 .org 0x00000168
362 .org 0x00000169
363 .org 0x0000016A
364 .org 0x0000016B
365 .org 0x0000016C
366 .org 0x0000016D
367 .org 0x0000016E
368 .org 0x0000016F
369 .org 0x00000170
370 .org 0x00000171
371 .org 0x00000172
372 .org 0x00000173
373 .org 0x00000174
374 .org 0x00000175
375 .org 0x00000176
376 .org 0x00000177
377 .org 0x00000178
378 .org 0x00000179
379 .org 0x0000017A
380 .org 0x0000017B
381 .org 0x0000017C
382 .org 0x0000017D
383 .org 0x0000017E
384 .org 0x0000017F
385 .org 0x00000180
```

Assembler

```
example.asm(4) : Error : Unknown mnemonic/macro: /*
```

- AVR Assembler
 - Doesn't support multi-line comments
- So I made an assembler
 - Supports multi-line comments
 - Not much else

Preprocess

- Strip out comments
- Look for labels
- Look for directives
- Store stuff in arrays

```
line = re.sub(r";.*|\s*\s*|\s*\s*\s*", "", line)
```

```
.include "m328Pdef.inc"
ldi r16,0b00100000
out DDRB,r16 ; Set data direction to out
out PortB,r16 ; Set pins high or low
Start:
rjmp Start ; Loop
```

program[] =

ldi r16,0b00100000
out ddrb,r16
out portb,r16
rjmp Start

labels[] =

Key	value
ddrb	0x04
portb	0x05
and tons of stuff from the .inc file	
start	4

Get instruction format

- Load immediate: `ldi Rd, K`
- Clear register: `clr Rd`

1110	KKKK	dddd	KKKK
------	------	------	------

0010	01dd	dddd	dddd
------	------	------	------

0010	01dd	dddd	dddd
------	------	------	------

0010	01dd	dddd	dddd
------	------	------	------

```
"ldi": {
  "format": "register-immediate",
  "opcode": "0b1110"
},
"register-immediate": {
  "regex": ".*\\s*(?P<Rd>\\w+),\\s*(?P<K>\\w+)",
  "fields": {
    "opcode": {
      "bits": [15,14,13,12]
    },
    "Rd": {
      "bits": [7,6,5,4]
    },
    "K": {
      "bits": [11,10,9,8,3,2,1,0]
    }
  }
}
```

Evaluate fields

- Store these values in an object
- Evaluate
 - Convert to integer
 - Evaluate general purpose register
 - Look up label



```
ldi r16,0b00100000

fieldValues = {
  opcode: "0b1110"
  Rd: "r16"
  K: "0b00100000"
}
```



```
fieldValues = {
  opcode: 14
  Rd: 16
  K: 32
}
```

Convert to bytecode

`ldi r16,0b00100000` converts to `0x00E2`

- Convert program to intel hex format

```
:0800000000E204B905B9FFCFCD
:00000001FF ↑
```

- Program data
- Data length (8 bytes)
- Address start (don't care)
- Record type (00 = data)
- Checksum (sum instructions, take twos complement, take last byte)
- EOF

Upload with AVRDUDE

- AVR Downloader/UploaDEr

- Command for Arduino Uno:

```
avrdude -p m328p -c arduino -P /dev/ttyUSB0 -b 115200 -U flash:w:led0n.hex
```

Demo

References

- Check it out on GitHub: <https://github.com/CalebJ2/avr-assembler>
- <https://www.microchip.com/webdoc/avrassembler/>
- https://en.wikipedia.org/wiki/Atmel AVR_instruction_set
- https://en.wikipedia.org/wiki/Intel_HEX
- <http://www.nongnu.org/avrdude/>
- <https://www.sparkfun.com/datasheets/Components/SMD/ATMega328.pdf>
- <https://github.com/lpodkalicki/blog/blob/master/avr/asm/include/m328Pdef.inc>
- <https://learn.sparkfun.com/tutorials/how-to-install-ch340-drivers/all>
- https://www.arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf
- <https://regex101.com>
- <https://pythonhosted.org/bitstring/>
- <https://www.instructables.com/id/Command-Line-AVR-Tutorials/>
- <https://www.codeproject.com/Articles/712610/AVR-Assembler>
- <http://nuft.github.io/avr/2015/08/02/avr-hex-programming.html>

Questions?