

CS 273

HW 8

(20) Z register is 2 bytes large.

Range: ~~\$0000~~ - ~~\$FFFF~~ (bytes)

bit 0: defines to read from low byte or high byte of the location

bits 1-15: define the address of the location

Location range = $2^{15} = 32,768$ locations

(22) Read high byte from $0x340$

- shift $0x340$ ~~right~~ left

- add 1

LDI ZL, \$81

LDI ZH, \$06

LPM r20, Z

$0x340 = 0000\ 0011\ 0100\ 0000$

0000 0110 1000 0001

$0x340$

$= \$06, \81

high byte

(24) Find y where $y = x^2 + 2x + 5$

; requires three registers: @0, @1, @2

; requires registers r0, r1 to be available

; $x = @3$, $0 \leq x \leq 9$, POSTCONDITION: @0 = y

.MACRO CALCULATE

LDI @0, @3

LDI @1, @3

MUL @0, @1

MOV @0, R0

LDI @3, Z

MUL @1, @2

ADD @0, R0

ADI @0, 5

.ENDMACRO

29) conversion:

```
.DB '0', '1', '2', '3', '4', '5', '6', '7',  
    '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'
```

```
LDI R31, hi8 (conversion)
```

```
LDI R30, lo8 (conversion)
```

```
LDI R20, $FO
```

```
OUT DDRB, R20
```

```
LDI R20, $FF
```

```
OUT DDRC, R20
```

```
IN R20, PORTB
```

```
INC R20
```

LOOP:

```
LPM R21, Z+
```

```
DEC R20
```

```
BRNE LOOP
```

```
OUT PORTC, R21
```

```
RET
```

64) Macro saves time, makes code look cleaner, less likely to have human error, can help save program memory, also you don't have to write it if you use someone else's.

65) Subroutine, because it uses the stack.