Notes:

 fugts: 2, 8, C, A, $p_1$, 0

ADD AL, [5x]
ADD AL, [3x]
ADD AL, TEMP
ADD AL, BL
ADD AL, 10

- SEC REST can be: Immediate, Registers,

REST $\rightarrow$ (SEC) + (REST)

- ADD REST, SRC

Addition

Arithmetic & Logic Unit

LeCunn 10
MUL CX, A
AX ← A * CX
MUL CL, A
AX ← A * CL

8-bit → 16-bit: 16-032: etc...

(rest) = AX/DX/Ax/EDx/EXx/kDX RX

MUL ZC

CMP 1
TEST, ZC

(Compare)

TEST (Comparator)

SBC 1
TEST, ZC

SUB (Test, ZC) = REST – (REST) – (SEC)

BL ← (BL) + (DL)

XADD BL, DL ← CL ← RL

(exchange Accumulator (ACD))

INC REST

REST ← INC
\[
\begin{aligned}
\text{sign}(x) \cdot \text{sign}(-x) &= 1 \\
\text{sign}(x) \cdot \text{sign}(x) &= \text{sign}(x)
\end{aligned}
\]

\[
\begin{aligned}
\text{AL} \text{, AL} \rightarrow \text{CL} \rightarrow \text{AL}
\end{aligned}
\]

\[
\begin{aligned}
\text{AL} - \text{AL} \\
\text{AL} - \text{AL} \\
\text{AL} - \text{AL} \\
\end{aligned}
\]

\[
\begin{aligned}
\text{uint} \text{ must be extended as a fixed-point number}
\end{aligned}
\]

\[
\begin{aligned}
\text{I\text{mul} ea integer multiplication - sign extends}
\end{aligned}
\]

\[
\begin{aligned}
0 - 0 \text{ if equal} \\
\text{and zero}
\end{aligned}
\]

\[
\begin{aligned}
\text{all meaningful} : C, \quad \text{higher order bits change but only if it's}
\end{aligned}
\]
NEC TEST
NOT RAS

BT C, TSC, RTS

BT RST, bit #

TEST RAS = 2

XOR, RAS = 2

OR, RAS = 2

AND, RAS = 2

Flags change; C & O not cleared.
(AX) not found or [SI] not found

Zero flag ≠ CX ≠ 0

CMPS comma string

SCAS scan string

BFR bit scan forward/reverse

BSEF bit scan forward

R0R ROR RRC

RCL RAL

REST, #bits (rotate)

MSB

SNF ≠ SAR

SAL = SAL (C) → 6

(SNIF)