

Lampiran A. Perhitungan teori dari hasil pengukuran benda bergerak jatuh bebas

Tabel A.1. Perhitungan teori percobaan 1

Mode	t_{Am} s	t_{Am} teori	t_{Bm} teori	t_{AB1} s	$t_{Bm} - t_{Am} = t_{ABm}$ teori
1	0.320	$t_{A1} = \sqrt{\frac{2 \times 0.50}{9.81}} = 0.319$	$t_{B1} = \sqrt{\frac{2 \times 1.00}{9.81}} = 0.451$	0.119	$0.451 - 0.319 = 0.132$
2	0.390	$t_{A2} = \sqrt{\frac{2 \times 0.75}{9.81}} = 0.391$	$t_{B2} = \sqrt{\frac{2 \times 1.50}{9.81}} = 0.553$	0.149	$0.553 - 0.391 = 0.162$
3	0.439	$t_{A3} = \sqrt{\frac{2 \times 1.00}{9.81}} = 0.451$	$t_{B3} = \sqrt{\frac{2 \times 2.00}{9.81}} = 0.638$	0.211	$0.638 - 0.451 = 0.187$

Tabel A.2. Perhitungan teori percobaan 2

Mode	t_{Am} s	t_{Am} teori	t_{Bm} teori	t_{AB1} s	$t_{Bm} - t_{Am} = t_{ABm}$ teori
1	318	$t_{A1} = \sqrt{\frac{2 \times 0.50}{9.81}} = 0.319$	$t_{B1} = \sqrt{\frac{2 \times 1.00}{9.81}} = 0.451$	120	$0.451 - 0.319 = 0.132$
2	388	$t_{A2} = \sqrt{\frac{2 \times 0.75}{9.81}} = 0.391$	$t_{B2} = \sqrt{\frac{2 \times 1.50}{9.81}} = 0.553$	151	$0.553 - 0.391 = 0.162$
3	438	$t_{A3} = \sqrt{\frac{2 \times 1.00}{9.81}} = 0.451$	$t_{B3} = \sqrt{\frac{2 \times 2.00}{9.81}} = 0.638$	211	$0.638 - 0.451 = 0.187$

Tabel A.3. Perhitungan teori percobaan 3

Mode	t_{Am} s	t_{Am} teori	t_{Bm} teori	t_{AB1} s	$t_{Bm} - t_{Am} = t_{ABm}$ teori
1	301	$t_{A1} = \sqrt{\frac{2 \times 0.50}{9.81}} = 0.319$	$t_{B1} = \sqrt{\frac{2 \times 1.00}{9.81}} = 0.451$	122	$0.451 - 0.319 = 0.132$
2	380	$t_{A2} = \sqrt{\frac{2 \times 0.75}{9.81}} = 0.391$	$t_{B2} = \sqrt{\frac{2 \times 1.50}{9.81}} = 0.553$	159	$0.553 - 0.391 = 0.162$
3	423	$t_{A3} = \sqrt{\frac{2 \times 1.00}{9.81}} = 0.451$	$t_{B3} = \sqrt{\frac{2 \times 2.00}{9.81}} = 0.638$	217	$0.638 - 0.451 = 0.187$

Lampiran B. Perhitungan ralat dari hasil pengukuran benda bergerak berubah beraturan.

Tabel B1. Data hasil percobaan.

Mode 1	t_{AM} ms	$\bar{v}_{AM}$ m/s	t_{ABM} ms	$\bar{v}_{BM}$ m/s	$\bar{a}_M$ m/s ²	$t_{AM} + t_{ABM}$
1	581	0.86	270	1.85	1.6376	851
2	558	0.87	279	1.79	1.6020	837
3	538	0.92	278	1.79	1.6109	816
4	545	0.91	279	1.79	1.6200	824
5	551	0.90	279	1.79	1.5931	830
Mode 2						
1	732	1.02	340	2.20	1.5546	1072
2	717	1.04	349	2.14	1.5051	1066
3	697	1.07	341	2.19	1.5840	1038
4	704	1.06	350	2.14	1.5199	1054
5	710	1.05	350	2.14	1.5051	1060
Mode 3						
1	851	1.17	422	2.36	1.3806	1273
2	837	1.19	447	2.23	1.2880	1284
3	816	1.22	432	2.31	1.3627	1248
4	824	1.21	448	2.23	1.2992	1272
5	830	1.20	440	2.27	1.3164	1270

Tabel B2. Perhitungan ralat t .

	$t_{AM} + t_{ABM} = t$ mili detik	$t_n - \bar{t} = \delta_n$ mili detik	δ^2	$s_t = \sqrt{\frac{\sum \delta^2}{5 \times 4}}$
Mode 1				
1	851	19.4	376.36	5.954830
2	837	5.4	29.16	
3	816	-15.6	243.36	
4	824	-7.6	57.76	
5	830	-1.6	2.56	
	$\bar{t} = 831.6$		$\sum \delta^2 = 709.2$	$t = (831.6 \pm 5.954830)$
				Ralat Relatif = 0.716069 %
Mode 2				
1	1072	14	196	5.830952
2	1066	8	64	
3	1038	-20	400	
4	1054	-4	16	
5	1060	2	4	
	$\bar{t} = 1058$		$\sum \delta^2 = 680$	$t = (1058 \pm 5.830952)$
				Ralat Relatif = 0.55113 %
Mode 3				
1	1273	3.6	12.96	5.878775
2	1284	14.6	213.16	
3	1248	-21.4	457.96	
4	1272	2.6	6.76	
5	1270	0.6	0.36	
	$\bar{t} = 1269.4$		$\sum \delta^2 = 691.2$	$t = (1269.4 \pm 5.878775)$
				Ralat Relatif = 0.463114 %

Tabel B3. Perhitungan ralat $\bar{v}_A$.

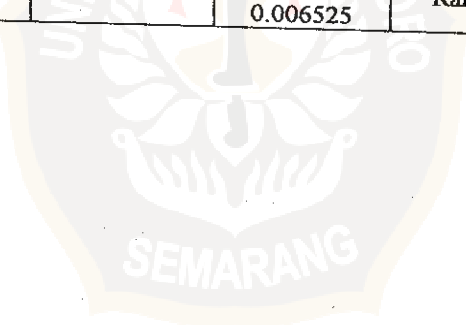
	$\bar{v}_{AM}$ meter/detik	$\bar{v}_n - \bar{v} = \delta_n$ meter/detik	δ^2	$s_i = \sqrt{\frac{\Sigma \delta^2}{5 \times 4}}$
Mode 1				
1	0.86	-0.032	0.001024	0.011575
2	0.87	-0.022	0.000484	
3	0.92	0.028	0.000784	
4	0.91	0.018	0.000324	
5	0.90	0.008	0.000064	$\bar{v}_A = (0.892 \pm 0.011575)$
	$\bar{v} = 0.892$		$\Sigma \delta^2 = 0.002680$	Ralat Relatif = 1.297646 %
Mode 2				
1	1.02	-0.028	0.000784	0.008602
2	1.04	-0.008	0.000064	
3	1.07	0.022	0.000484	
4	1.06	0.012	0.000144	
5	1.05	0.002	0.000004	$\bar{v}_A = (1.048 \pm 0.008602)$
	$\bar{v} = 1.048$		$\Sigma \delta^2 = 0.001480$	Ralat Relatif = 0.820802 %
Mode 3				
1	1.17	-0.028	0.000784	0.008602
2	1.19	-0.008	0.000064	
3	1.22	0.022	0.000484	
4	1.21	0.012	0.000144	
5	1.20	0.002	0.000004	$\bar{v}_A = (1.198 \pm 0.008602)$
	$\bar{v} = 1.198$		$\Sigma \delta^2 = 0.001480$	Ralat Relatif = 0.71803 %

Tabel B4. Perhitungan ralat $\bar{v}_B$.

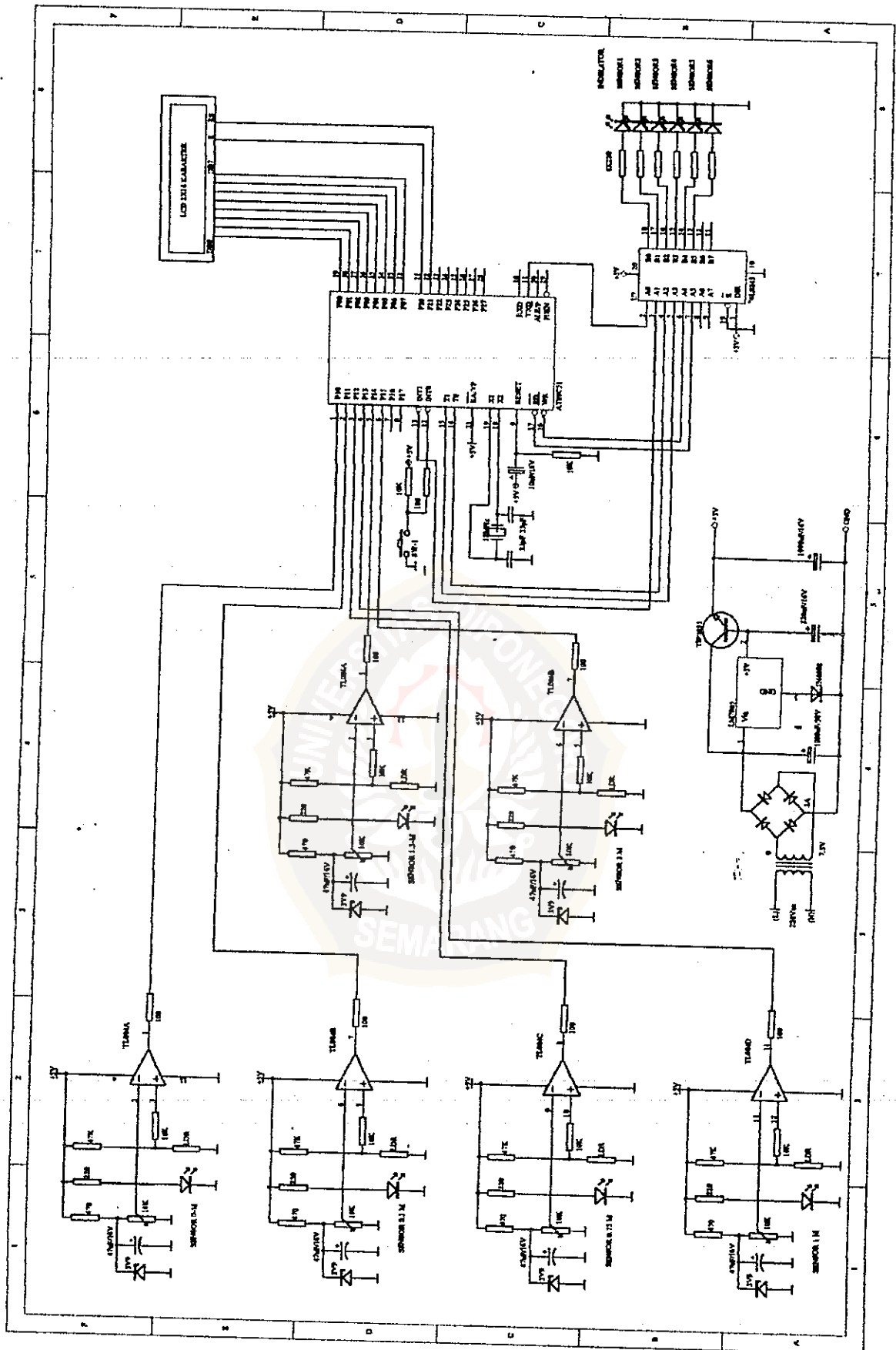
	$\bar{v}_{BM}$ meter/detik	$\bar{v}_n - \bar{v} = \delta_n$ meter/detik	δ^2	$s_i = \sqrt{\frac{\Sigma \delta^2}{5 \times 4}}$
Mode 1				
1	1.85	0.048	0.002304	0.011575
2	1.79	-0.012	0.000144	
3	1.79	-0.012	0.000144	
4	1.79	-0.012	0.000144	
5	1.79	-0.012	0.000144	$\bar{v}_B = (1.802 \pm 0.011575)$
	$\bar{v} = 1.802$		$\Sigma \delta^2 = 0.002880$	Ralat Relatif = 0.642342 %
Mode 2				
1	2.20	0.038	0.002304	0.008602
2	2.14	-0.022	0.000144	
3	2.19	0.028	0.000144	
4	2.14	-0.022	0.000144	
5	2.14	-0.022	0.000144	$\bar{v}_B = (2.162 \pm 0.008602)$
	$\bar{v} = 2.162$		$\Sigma \delta^2 = 0.003680$	Ralat Relatif = 0.397872 %
Mode 3				
1	2.20	0.08	0.006400	0.008602
2	2.14	-0.05	0.002500	
3	2.19	0.03	0.000900	
4	2.14	-0.05	0.002500	
5	2.14	-0.01	0.000100	$\bar{v}_B = (2.280 \pm 0.008602)$
	$\bar{v} = 2.280$		$\Sigma \delta^2 = 0.012400$	Ralat Relatif = 0.377281 %

Tabel B5. Perhitungan ralat $\bar{a}$.

	$\bar{a}_M$ meter/detik ²	$\bar{a}_n - \bar{\bar{a}} = \delta_n$ meter/detik ²	δ^2	$s_i = \sqrt{\frac{\Sigma \delta^2}{5 \times 4}}$
Mode 1				
1	1.6376	0.02488	0.000619	0.007665 $\bar{a} = (1.61272 \pm 0.007665)$
2	1.6020	-0.01072	0.000115	
3	1.6109	-0.00182	0.000003	
4	1.6200	0.00728	0.000053	
5	1.5931	-0.01962	0.000385	
	$\bar{\bar{a}} = 1.61272$		$\Sigma \delta^2 =$ 0.001175	Ralat Relatif = 0.475284 %
Mode 2				
1	1.5546	0.02086	0.000435	0.015481 $\bar{a} = (1.53374 \pm 0.015481)$
2	1.5051	-0.02864	0.000820	
3	1.5840	0.05026	0.002526	
4	1.5199	-0.01384	0.000192	
5	1.5051	-0.02864	0.000820	
	$\bar{\bar{a}} = 1.53374$		$\Sigma \delta^2 =$ 0.004793	Ralat Relatif = 1.009363 %
Mode 3				
1	1.3806	0.05122	0.002623	0.018063 $\bar{a} = (1.32938 \pm 0.018063)$
2	1.2880	-0.04138	0.001712	
3	1.3627	0.03332	0.001110	
4	1.2992	-0.03018	0.000911	
5	1.3164	-0.01298	0.000168	
	$\bar{\bar{a}} = 1.32938$		$\Sigma \delta^2 =$ 0.006525	Ralat Relatif = 1.358754 %



Lampiran C.1 Rangkaian Alat.



Lampiran C2. Daftar Komponen

DAFTAR KOMPONEN SENSOR CAHAYA

NO	NAMA	SPESIFIKASI	N	KETERANGAN
1	RESISTOR	1000 HM	6	
2	RESISTOR	220 OHM	6	
3	RESISTOR	470 OHM	6	
4	RESISTOR	10 KOHM	6	
5	RESISTOR	47 KOHM	6	
6	VARIABEL RESISTOR	10 KOHM	6	
7	LDR	20 KOHM (MIN)	6	
8	KAPASITOR ELEKTOLIT	47 uF/16V	6	
9	DIODA ZENER	3V9 0.25W	6	
10	LED BENING	MERAH	6	
11	IC OPAMP	CA 3140	6	
12	SOKET IC	8 PIN	6	

DAFTAR KOMPONEN SISTEM AT89C51

NO	NAMA	SPESIFIKASI	N	KETERANGAN
1	RESISTOR	100 OHM	6	
2	RESISTOR	10 KOHM	6	
3	KAPASITOR ELEKTOLIT	10uF/16V	1	
4	KAPASITOR KERAMIK	33pF	2	
5	KRISTAL	12MHZ	1	
6	IC MIKROKONTROLLER	AT89C51	1	
7	SOKET IC	40 PIN	1	
8	SAKLAR	PUSH BUTTON		

DAFTAR KOMPONEN BUFFER LED INDIKATOR

NO	NAMA	SPESIFIKASI	N	KETERANGAN
1	RESISTOR	100 OHM	6	
2	IC BUFFER	75LS245	1	
3	SOKET IC	20 PIN	1	
4	LED	HIJAU	6	

DAFTAR KOMPONEN CATU DAYA

NO	NAMA	SPESIFIKASI	N	KETERANGAN
1	DIODA JEMBATAN	3A	1	
2	DIODA	1N4002	1	
3	KAPASITOR ELEKTROLIT	1000uF/50V	1	
4	KAPASITOR ELEKTROLIT	1000uF/16V	1	
5	KAPASITOR ELEKTROLIT	220uF/16V	1	
6	IC REGULATOR	LM7805	1	
7	TRANSISTOR	TIP 3055	1	
8	TRANSORMATOR	220VAC/7.5VAC 1A	1	

DAFTAR KOMPONEN LCD

NO	NAMA	SPESIFIKASI	N	KETERANGAN
1	LCD 2X16 KARAKTER	TM126A	1	

Lampiran D. Listing Program yang dipakai.

```

; VELOCIT2.ASM
X12 EQU 00H ; jarak 0.5 m = 500 mm
X11 EQU 0C3H ; untuk pembagian 2 di belakang koma, maka
X10 EQU 50H ; 500 x 100 = 50000 -> C350(H)

X22 EQU 01H ; jarak 0.75 m = 750 mm
X21 EQU 24H ; untuk pembagian 2 di belakang koma, maka
X20 EQU 0F8H ; 750 x 100 = 75000 -> 0124F8(H)

X32 EQU 01H ; jarak 1 m = 1000 mm
X31 EQU 86H ; untuk pembagian 2 di belakang koma, maka
X30 EQU 0A0H ; 1000 x 100 = 100000 -> 0186A0H

CLOCK0 EQU 20H ; menghitung tiap 250 mikro detik
TMDTK1 EQU 21H ; variabel mili detik byte-1
TMDTK0 EQU 22H ; variabel mili detik byte-0

MDT1A1 EQU 23H ; waktu (byte-1) di titik A posisi 1
MDT1A0 EQU 24H ; waktu (byte-0) di titik A posisi 1
MDT1B1 EQU 25H ; waktu (byte-1) di titik B posisi 1
MDT1B0 EQU 26H ; waktu (byte-0) di titik B posisi 1

MDT2A1 EQU 27H ; waktu (byte-1) di titik A posisi 2
MDT2A0 EQU 28H ; waktu (byte-0) di titik A posisi 2
MDT2B1 EQU 29H ; waktu (byte-1) di titik B posisi 2
MDT2B0 EQU 2AH ; waktu (byte-0) di titik B posisi 2

MDT3A1 EQU 2BH ; waktu (byte-1) di titik A posisi 3
MDT3A0 EQU 2CH ; waktu (byte-0) di titik A posisi 3
MDT3B1 EQU 2DH ; waktu (byte-1) di titik B posisi 3
MDT3B0 EQU 2EH ; waktu (byte-0) di titik B posisi 3

FLGERR EQU 2FH ; deteksi error (over time)
FLGAWL EQU 30H ; mulai ke awal

DATA EQU 31H ; data LCD
POSX EQU 32H ; posisi kolom LCD

INDEX EQU 33H ; variabel index
BARIS EQU 34H ; posisi baris LCD

RATRIB EQU 35H ; nilai ratusan ribu
PULRIB EQU 36H ; nilai puluhan ribu
RIBUAN EQU 37H ; nilai ribuan
RATUS EQU 38H ; nilai ratusan
PULUH EQU 39H ; nilai puluhan
SATUAN EQU 3AH ; nilai satuan

OP3 EQU 3BH ; variabel-1 aritmatika byte-3
OP2 EQU 3CH ; variabel-1 aritmatika byte-2
OP1 EQU 3DH ; variabel-1 aritmatika byte-1
OP0 EQU 3EH ; variabel-1 aritmatika byte-0

ANGKA2 EQU 3FH ; variabel-2 aritmatika byte-2
ANGKA1 EQU 40H ; variabel-2 aritmatika byte-1
ANGKA0 EQU 41H ; variabel-2 aritmatika byte-0

TMP3 EQU 42H ; variabel penyimpan sementara byte-3
TMP2 EQU 43H ; variabel penyimpan sementara byte-2
TMP1 EQU 44H ; variabel penyimpan sementara byte-1
TMP0 EQU 45H ; variabel penyimpan sementara byte-0

SISA1 EQU 46H ; variabel sisa hasil bagi byte-1
SISA0 EQU 47H ; variabel sisa hasil bagi byte-0

VIA1 EQU 48H ; kecepatan (byte-1) di titik A posisi 1

```


V1A0	EQU	49H	; kecepatan (byte-0) di titik A posisi 1
V1B1	EQU	4AH	; kecepatan (byte-1) di titik B posisi 1
V1B0	EQU	4BH	; kecepatan (byte-0) di titik B posisi 1
T1AB1	EQU	4CH	; waktu (byte-1) di titik AB posisi 1
T1AB0	EQU	4DH	; waktu (byte-0) di titik AB posisi 1
ACCL12	EQU	4EH	; percepatan (byte-2) di posisi 1
ACCL11	EQU	4FH	; percepatan (byte-1) di posisi 1
ACCL10	EQU	50H	; percepatan (byte-0) di posisi 1
V2A1	EQU	51H	; kecepatan (byte-1) di titik A posisi 2
V2A0	EQU	52H	; kecepatan (byte-0) di titik A posisi 2
V2B1	EQU	53H	; kecepatan (byte-1) di titik B posisi 2
V2B0	EQU	54H	; kecepatan (byte-0) di titik B posisi 2
T2AB1	EQU	55H	; waktu (byte-1) di titik AB posisi 2
T2AB0	EQU	56H	; waktu (byte-0) di titik AB posisi 2
ACCL22	EQU	57H	; percepatan (byte-2) di posisi 2
ACCL21	EQU	58H	; percepatan (byte-1) di posisi 2
ACCL20	EQU	59H	; percepatan (byte-0) di posisi 2
V3A1	EQU	5AH	; kecepatan (byte-1) di titik A posisi 3
V3A0	EQU	5BH	; kecepatan (byte-0) di titik A posisi 3
V3B1	EQU	5CH	; kecepatan (byte-1) di titik B posisi 3
V3B0	EQU	5DH	; kecepatan (byte-0) di titik B posisi 3
T3AB1	EQU	5EH	; waktu (byte-1) di titik AB posisi 3
T3AB0	EQU	5FH	; waktu (byte-0) di titik AB posisi 3
ACCL32	EQU	60H	; percepatan (byte-2) di posisi 3
ACCL31	EQU	61H	; percepatan (byte-1) di posisi 3
ACCL30	EQU	62H	; percepatan (byte-0) di posisi 3
BUFFRC	EQU	63H	; penyimpan register C saat interrupt
KAR1	EQU	64H	; penyimpan nilai Karakter-1
KAR2	EQU	65H	; penyimpan nilai Karakter-2
KAR3	EQU	66H	; penyimpan nilai Karakter-3
KAR4	EQU	67H	; penyimpan nilai Karakter-4
KAR5	EQU	68H	; penyimpan nilai Karakter-5
KAR6	EQU	69H	; penyimpan nilai Karakter-6
KRSIGN	EQU	6AH	; penyimpan tanda (-) untuk percepatan
KSIGN1	EQU	6BH	; penyimpan tanda (-) untuk percepatan-1
KSIGN2	EQU	6CH	; penyimpan tanda (-) untuk percepatan-2
KSIGN3	EQU	6DH	; penyimpan tanda (-) untuk percepatan-3
BUF2	EQU	6EH	; variabel penyimpan sementara
BUF1	EQU	6FH	; variabel penyimpan sementara
BUF0	EQU	70H	; variabel penyimpan sementara
BUF5	EQU	71H	; variabel penyimpan sementara
BUF6	EQU	72H	; variabel penyimpan sementara
BUF7	EQU	73H	; variabel penyimpan sementara
V1AR1	EQU	74H	; kecepatan rata2 (byte-1) di titik A posisi 1
V1AR0	EQU	75H	; kecepatan rata2 (byte-0) di titik A posisi 1
V1BR1	EQU	76H	; kecepatan rata2 (byte-1) di titik B posisi 1
V1BR0	EQU	77H	; kecepatan rata2 (byte-0) di titik B posisi 1
V2AR1	EQU	78H	; kecepatan rata2 (byte-1) di titik A posisi 2
V2AR0	EQU	79H	; kecepatan rata2 (byte-0) di titik A posisi 2
V2BR1	EQU	7AH	; kecepatan rata2 (byte-1) di titik B posisi 2
V2BR0	EQU	7BH	; kecepatan rata2 (byte-0) di titik B posisi 2
V3AR1	EQU	7CH	; kecepatan rata2 (byte-1) di titik A posisi 3
V3AR0	EQU	7DH	; kecepatan rata2 (byte-0) di titik A posisi 3
V3BR1	EQU	7EH	; kecepatan rata2 (byte-1) di titik B posisi 3
V3BR0	EQU	7FH	; kecepatan rata2 (byte-0) di titik B posisi 3


```

; PROGRAM VELOCITI1.ASM
; V MINIMAL 0.10 m/dt --> t MAKSIMAL (2m) : 20000 mdt (4E20H)
; V MAKSIMAL 100.00 m/dt --> t MINIMAL (0.5m) : 5 mdt (0005H)
;
; X MINIMAL 0.5 m = 500 mm
; untuk pembagian dengan hasil 2 angka di belakang koma, maka
; dipakai angka 50000 (C350H)
;
; X MAKSIMAL 2 m = 2000 mm
; untuk pembagian dengan hasil 2 angka di belakang koma, maka
; dipakai angka 200000 (030D40H)
;
; FUNGSI ARITMATIKA YANG DIPAKAI
; PERKALIAN 15 BIT (MULT16)
; PEMBAGIAN 24 BIT (DIV24)
; PENGURANGAN 16 BIT (SUBB24)
; PENAMBAHAN 24 BIT (ADD24)
;
; INTO : P3.2 --> INTERRUPT UNTUK MULAI LAGI
;
      ORG 0000H ; alamat awal-reset
      AJMP START
; alamat interrupt INTO
      ORG 0003H
      LJMP RSTAWL
; alamat interrupt TIMERO
      ORG 000BH
      LJMP .TIMERO
; alamat program utama
      ORG 0100H
START: LCALL DELON ; delay on
      LCALL INISIA ; inisialisasi LCD
      LCALL DSPCLR ; hapus tampilan LCD
      LCALL CURSOF ; cursor off

      MOV TMOD,#00100010B ; timer 1/0 mode 2
      MOV TH0,#06 ; nilai untuk timer 250 mikro detik
      MOV TL0,#06 ;
      MOV IE,#83H ; INTERRUPT TO DAN INTO
      SETB IT0 ; FALLING EDGE INTO

AWAL: LCALL TMPAWL ; tampilan awal
      MOV CLOK0,#00 ; inisialisasi variabel
      MOV TMDTK1,#00
      MOV TMDTK0,#00
      MOV FLGERR,#00
      MOV FLGAWL,#00

      CLR P3.1 ; reset indikator sensor-1
      CLR P3.3 ; reset indikator sensor-2
      CLR P3.4 ; reset indikator sensor-3
      CLR P3.5 ; reset indikator sensor-4
      CLR P3.6 ; reset indikator sensor-5
      CLR P3.7 ; reset indikator sensor-6

      JB P1.0,$ ; tunggu sampai sensor-1 on (0 m)
      SETB TR0 ; start timer-0
      SETB P3.1 ; indikator sensor-1 on

      UIAP11: JB P1.1,CKERR1 ; tunggu sampai sensor-2 on (0.5 m)
              SETB P3.3 ; indikator sensor-2 on
              LCALL BCWKU ; baca waktu
              MOV MDT1A1,A ; simpan waktu byte-1 (titik A posisi 1)
              MOV MDT1A0,B ; simpan waktu byte-0 (titik A posisi 1)

      UIAP12: JB P1.2,CKERR2 ; tunggu sampai sensor-3 on (0.75 m)
              SETB P3.4 ; indikator sensor-3 on
              LCALL BCWKU ; baca waktu
              MOV MDT2A1,A ; simpan waktu byte-1 (titik A posisi 2)
              MOV MDT2A0,B ; simpan waktu byte-0 (titik A posisi 2)

```

```

ULAP13: JB      P1.3,CKERR3      ; tunggu sampai sensor-4 on (1.0 m)
        SETB    P3.5             ; indikator sensor-4 on
        LCALL   BCWKTU           ; baca waktu
        MOV     MDT1B1,A         ; simpan waktu byte-1 (titik B posisi 1)
        MOV     MDT1B0,B         ; simpan waktu byte-0 (titik B posisi 1)
        MOV     MDT3A1,A         ; simpan waktu byte-1 (titik A posisi 3)
        MOV     MDT3A0,B         ; simpan waktu byte-0 (titik A posisi 3)

ULAP14: JB      P1.4,CKERR4      ; tunggu sampai sensor-5 on (1.5 m)
        SETB    P3.6             ; indikator sensor-5 on
        LCALL   BCWKTU           ; baca waktu
        MOV     MDT2B1,A         ; simpan waktu byte-1 (titik B posisi 2)
        MOV     MDT2B0,B         ; simpan waktu byte-0 (titik B posisi 2)

ULAP15: JB      P1.5,CKERR5      ; tunggu sampai sensor-6 on (2.0 m)
        CLR     TRO              ; stop timer
        SETB    P3.7             ; indikator sensor-6 on
        LCALL   BCWKTU           ; baca waktu
        MOV     MDT3B1,A         ; simpan waktu byte-1 (titik B posisi 3)
        MOV     MDT3B0,B         ; simpan waktu byte-0 (titik B posisi 3)

TESX:  LCALL   HITUNG            ; panggil proses perhitungan

ULTMPL: LCALL   TMPHSL           ; panggil tampilan hasil
        MOV     A,FLGAWL         ; deteksi mulai dari awal
        MOV     B,#01
        CJNE    A,B,JPEXIT
        LJMP     A,B,ULTMPL
        AWAL

;
; Cek error (over time)
;
CKERR1: MOV     A,FLGERR
        MOV     B,#00
        CJNE    A,B,JPEXIT
        LJMP     ULAP11

CKERR2: MOV     A,FLGERR
        MOV     B,#00
        CJNE    A,B,JPEXIT
        LJMP     ULAP12

CKERR3: MOV     A,FLGERR
        MOV     B,#00
        CJNE    A,B,JPEXIT
        LJMP     ULAP13

CKERR4: MOV     A,FLGERR
        MOV     B,#00
        CJNE    A,B,JPEXIT
        LJMP     ULAP14

CKERR5: MOV     A,FLGERR
        MOV     B,#00
        CJNE    A,B,JPEXIT
        LJMP     ULAP15

JPEXIT: CLR     TRO
        LCALL   TMPERR           ; Tampil ERROR, MULAI DARI AWAL
        LJMP     AWAL

HITUNG: NOP
        LCALL   HITPO1           ; panggil hitungan posisi-1
        LCALL   HITPO2           ; panggil hitungan posisi-2
        LCALL   HITPO3           ; panggil hitungan posisi-3
        RET

; *** PERHITUNGAN DI POSISI 1 ***
;
; POSISI 1 : X1A = 0.5 M --> 50000 (C350H)

```



```

;
;           XiB = 1 M --> 100000 (186A0H)
;
HITP01: LCALL HITV1A      ; panggil perhitungan kec. A posisi-1
        LCALL HIT1AB      ; panggil perhitungan waktu AB posisi-1
        LCALL HITV1B      ; panggil perhitungan kec. B posisi-1
        LCALL HITAC1      ; panggil perhitungan percepatan posisi-1
        RET

;
; HITUNG KECEPATAN DI POSISI 1 TITIK A
;
HITV1A: MOV     OP3,#00H   ; Jarak 0.5 m,
        MOV     OP2,#X12   ; (50000) = 00 C3 50 (H)
        MOV     OP1,#X11   ;
        MOV     OP0,#X10   ;
        MOV     ANGKA1,MDT1A1 ; waktu di titik A posisi-1
        MOV     ANGKA0,MDT1A0 ;
        LCALL    DIV24      ; jarak / waktu
        MOV     V1A1,OP1    ; simpan hasil pembagian di
        MOV     V1A0,OP0    ; V1A1 V1A0 (16 bit)

;
; HITUNG KECEPATAN RATA-RATA DI TITIK 0 - A POSISI-1
;
        MOV     OP3,#00     ; ambil nilai V1A1 V1A0
        MOV     OP2,#00
        MOV     ANGKA1,#00   ; pembagi = 2
        MOV     ANGKA0,#02
        LCALL    DIV24      ; (V1A1 V1A0) / 2
        MOV     V1A1,OP1    ; simpan hasil pembagian di
        MOV     V1A0,OP0    ; V1A1 V1A0 (16 bit)
        RET

;
; HITUNG T1AB
;
HIT1AB: MOV     OP2,#00     ; ambil waktu di titik B posisi-1
        MOV     OP1,MDT1B1   ; MDT1B1 MDT1B0
        MOV     OP0,MDT1B0
        MOV     ANGKA2,#00   ; ambil waktu di titik A posisi-1
        MOV     ANGKA1,MDT1A1 ; MDT1A1 MDT1A0
        MOV     ANGKA0,MDT1A0
        LCALL    SUBB24      ; (MDT1B1 MDT1B0) - (MDT1A1 MDT1A0)
        MOV     T1AB1,OP1    ; simpan hasil pengurangan di
        MOV     T1AB0,OP0    ; T1AB1 T1AB0 (16 BIT)
        RET

;
; HITUNG KECEPATAN DI POSISI 1 TITIK B
;
HITV1B: MOV     OP3,#00H   ; Jarak 0.5 m
        MOV     OP2,#X12   ;
        MOV     OP1,#X11   ;
        MOV     OP0,#X10   ;
        MOV     ANGKA1,T1AB1 ; waktu antara titik A - B posisi 1
        MOV     ANGKA0,T1AB0 ;
        LCALL    DIV24      ; jarak / waktu
        MOV     V1B1,OP1    ; simpan hasil pembagian di
        MOV     V1B0,OP0    ; V1B1 V1B0

;
; HITUNG KECEPATAN RATA-RATA DI TITIK A-B POSISI-1
;
        MOV     OP2,#00
        MOV     ANGKA2,#00   ; Ambil V1A1 V1A0
        MOV     ANGKA1,V1A1
        MOV     ANGKA0,V1A0
        LCALL    ADD24      ; (V1B1 V1B0) + (V1A1 V1A0)

        MOV     OP3,#00
        MOV     ANGKA2,#00   ; PEMBAGI = 2
        MOV     ANGKA1,#00
        MOV     ANGKA0,#02
        LCALL    DIV24      ; [(V1B1 V1B0) + (V1A1 V1A0)]/2

```

```

MOV     V1BR1,OP1      ; Simpan hasil pembagian di
MOV     V1BR0,OP0      ; V1BR1 V1BR0
RET

;
; HITUNG PERCEPATAN DI POSISI 1 TITIK B
;
HITAC1: NOP
;
; HITUNG VIA * V1A
;
MOV     OP2,#00
MOV     OP1,V1AR1
MOV     OP0,V1AR0
MOV     R3,V1AR1
MOV     R2,V1AR0
DEC     R2
LCALL   MULT16          ; (V1AR1 V1AR0) * (V1AR1 V1AR0)
MOV     BUF2,OP2        ; simpan hasil perkalian di
MOV     BUF1,OP1        ; BUF2 BUF1 BUF0 (24 bit)
MOV     BUF0,OP0

;
; HITUNG V1B * V1B
;
MOV     OP2,#00
MOV     OP1,V1BR1
MOV     OP0,V1BR0
MOV     R3,V1BR1
MOV     R2,V1BR0
DEC     R2
LCALL   MULT16          ; (V1BR1 V1BR0) * (V1BR1 V1BR0)
MOV     BUF7,OP2        ; simpan hasil pembagian di
MOV     BUF6,OP1        ; BUF7 BUF6 BUF5 (24 bit)
MOV     BUF5,OP0

;
; HITUNG V1B*V1B - V1A*V1A
;
CLR     C
MOV     KSIGN1,#20H
MOV     A,V1B1
MOV     B,V1A1
CJNE    A,B,J1BSR1      ; cek apakah V1B1 = V1A1
MOV     A,V1B0
MOV     B,V1A0
CJNE    A,B,J1BSR1      ; cek apakah V1B0 = V1A0
MOV     ACC112,#00      ; bila sama, maka percepatan = 0
MOV     ACC111,#00
MOV     ACC110,#00
RET

J1BSR1: JNC     J1BES1
;
; PERCEPATAN NEGATIF
; (V1AR1 V1AR0)^2 - (V1BR1 V1BR0)^2
;
MOV     KSIGN1,#2DH      ; TANDA '-'
MOV     OP2,BUF2
MOV     OP1,BUF1
MOV     OP0,BUF0
MOV     ANGKA2,BUF7
MOV     ANGKA1,BUF6
MOV     ANGKA0,BUF5
LJMP    J1SB24
;
; PERCEPATAN POSITIF
; (V1BR1 V1BR0)^2 - (V1AR1 V1AR0)^2
;
J1BES1: MOV     OP2,BUF7
MOV     OP1,BUF6
MOV     OP0,BUF5
MOV     ANGKA2,BUF2

```

```

MOV     ANGKA1,BUF1
MOV     ANGKA0,BUF0

J1SB24: LCALL  SUBB24
;
; (V1B^2-V1A^2) / (2*X)
; 2*X = 2*0.5 m = 1 m
; a = (V1B^2-V1A^2)      (4 DIGIT DI BELAKANG KOMA)
MOV     ACCL12,OP2      ; simpan hasil di
MOV     ACCL11,OP1      ; ACCL12 ACCL11 ACCL10 (24 bit)
MOV     ACCL10,OP0
RET

; *** PERHITUNGAN DI POSISI 2
;
; POSISI 2 : X2A = 0.75 M --> 75000 (124F8H)
;           T2A min = 7.5 mdtk
;           T2A max = 7500 mdtk
;
;           X2B = 1.5 M --> 150000 (249F0H)
;           T2B min = 15 mdtk
;           T2B max = 15000 mdtk
HITPO2: LCALL  HITV2A      ; panggil perhitungan kec. A posisi-2
        LCALL  HIT2AB      ; panggil perhitungan waktu AB posisi-2
        LCALL  HITV2B      ; panggil perhitungan kec. B posisi-2
        LCALL  HITAC2      ; panggil perhitungan percepatan posisi-2
        RET

;
; HITUNG KECEPATAN DI POSISI 2 TITIK A
;
HITV2A: MOV     OP3,#00H      ; Jarak 0.75 m
        MOV     OP2,#X22      ; (75000 = 1 24 F8 H)
        MOV     OP1,#X21      ;
        MOV     OP0,#X20      ;
        MOV     ANGKA1,MDT2A1 ; waktu di titik A posisi-2
        MOV     ANGKA0,MDT2A0 ;
        LCALL  DIV24          ; jarak / waktu
        MOV     V2A1,OP1      ; simpan hasil pembagian di
        MOV     V2A0,OP0      ; V2A1 V2A0 (16 bit)
;
; HITUNG KECEPATAN RATA-RATA DI TITIK 0 - A POSISI-2
;
        MOV     OP3,#00
        MOV     OP2,#00
        MOV     ANGKA1,#00    ; pembagi = 2
        MOV     ANGKA0,#02
        LCALL  DIV24          ; (V2A1 V2A0) / 2
        MOV     V2A1,OP1      ; Simpan hasil pembagian di
        MOV     V2A0,OP0      ; V2A1 V2A0 (16 bit)
        RET

;
; HITUNG T2AB
;
HIT2AB: MOV     OP2,#00      ; ambil waktu di titik B posisi-2
        MOV     OP1,MDT2B1    ; MDT2B1 MDT2B0
        MOV     OP0,MDT2B0
        MOV     ANGKA2,#00    ; ambil waktu di titik A posisi-2
        MOV     ANGKA1,MDT2A1 ; MDT2A1 MDT2A0
        MOV     ANGKA0,MDT2A0
        LCALL  SUBB24          ; (MDT2B1 MDT2B0) - (MDT2A1 MDT2A0)
        MOV     T2AB1,OP1     ; simpan hasil pengurangan di
        MOV     T2AB0,OP0     ; T2AB1 T2AB0 ( 16 bit)
        RET

;
; HITUNG KECEPATAN DI POSISI 2 TITIK B
;
HITV2B: MOV     OP3,#00H      ; Jarak 0.75 m
        MOV     OP2,#X22      ; (75000 = 1 24 F8 H)

```

```

MOV     OP1,#X21      ;
MOV     OP0,#X20      ;
MOV     ANGKA1,T2AB1  ; ambil waktu antara titik A-B posisi-2
MOV     ANGKA0,T2AB0  ; T2AB1 T2AB0
LCALL   DIV24         ; jarak / waktu
MOV     V2B1,OP1      ; simpan hasil pembagian di
MOV     V2B0,OP0      ; V2B1 V2B0 (16 bit)
;
; HITUNG KECEPATAN RATA-RATA DI TITIK A-B POSISI-2
;
MOV     OP2,#00
MOV     ANGKA2,#00    ; ambil kecepatan di titik A posisi-2
MOV     ANGKA1,V2A1    ; V2A1 V2A0
MOV     ANGKA0,V2A0
LCALL   ADD24         ; (V2AB1 V2AB0) + (V2A1 V2A0)

MOV     OP3,#00
MOV     ANGKA2,#00    ; pembagi = 2
MOV     ANGKA1,#00
MOV     ANGKA0,#02
LCALL   DIV24         ; [(V2AB1 V2AB0) + (V2A1 V2A0)]/2
MOV     V2BR1,OP1     ; simpan hasil pembagian di
MOV     V2BR0,OP0     ; V2BR1 V2BR0
RET

;
; HITUNG PERCEPATAN DI POSISI 1 TITIK B
;
HITAC2: NOP
;
; HITUNG V2A * V2A
;
MOV     OP2,#00
MOV     OP1,V2AR1
MOV     OP0,V2AR0
MOV     R3,V2AR1
MOV     R2,V2AR0
DEC     R2
LCALL   MULT16        ; (V2AR1 V2AR0) * (V2AR1 V2AR0)
MOV     BUF2,OP2      ; simpan hasil perkalian di
MOV     BUF1,OP1      ; BUF2 BUF1 BUF0 (24 bit)
MOV     BUF0,OP0
;
; HITUNG V2B * V2B
;
MOV     OP2,#00
MOV     OP1,V2BR1
MOV     OP0,V2BR0
MOV     R3,V2BR1
MOV     R2,V2BR0
DEC     R2
LCALL   MULT16        ; (V2BR1 V2BR0) * (V2BR1 V2BR0)
MOV     BUF7,OP2      ; simpan hasil perkalian di
MOV     BUF6,OP1      ; BUF7 BUF6 BUF5 (24 bit)
MOV     BUF5,OP0
;
; HITUNG V2B*V2B - V2A*V2A
;
CLR     C
MOV     KSIGN2,#20H
MOV     A,V2B1
MOV     B,V2A1
CJNE   A,B,J2BSR1     ; cek apakah V2B1 = V2A1
MOV     A,V2B0
MOV     B,V2A0
CJNE   A,B,J2BSR1     ; cek apakah V2B0 = V2A0
MOV     ACCL22,#00    ; jika sama, maka
MOV     ACCL21,#00    ; percepatan = 0
MOV     ACCL20,#00
RET

```

```

J2BSR1: JNC     J2BES1
;
; PERCEPATAN NEGATIF
; (V2AR1 V2AR0)^2 - (V2BR1 V2BR0)^2
;
      MOV     KSIGN2,#2DH      ; TANDA '-'
      MOV     OP2,BUF2
      MOV     OP1,BUF1
      MOV     OP0,BUF0
      MOV     ANGKA2,BUF7
      MOV     ANGKA1,BUF6
      MOV     ANGKA0,BUF5
      JMP     J2SB24

;
; PERCEPATAN POSITIF
; (V2AR1 V2AR0)^2 - (V2BR1 V2BR0)^2
J2BES1: MOV     OP2,BUF7
      MOV     OP1,BUF6
      MOV     OP0,BUF5
      MOV     ANGKA2,BUF2
      MOV     ANGKA1,BUF1
      MOV     ANGKA0,BUF0

J2SB24: LCALL    SUBB24
;
; (V2B^2-V2A^2) / (2*X)
; 2*X = 2*0.75 m = 1.5 m
; a = (V1B^2-V1A^2)*10 / 15      (4 DIGIT DI BELAKANG KOMA)
      MOV     R3,#00          ; pengali = 10
      MOV     R2,#10
      DEC     R2
      LCALL    MULT16          ; (V1B^2-V1A^2)*10
      MOV     ANGKA2,#00      ; pembagi = 15
      MOV     ANGKA1,#00
      MOV     ANGKA0,#15
      LCALL    DIV24           ; (V1B^2-V1A^2)*10 / 15
      MOV     ACCL22,OP2      ; simpan hasil pembagian di
      MOV     ACCL21,OP1      ; ACCL22 ACCL21 ACCL20 (24 bit)
      MOV     ACCL20,OP0
      RET

; *** PERHITUNGAN DI POSISI 3
;
; POSISI 3 : X3A = 1 M --> 100000 (186A0H)
;           T3A min = 10 mdtk
;           T3A max = 10000 mdtk
;
;           X3B = 2 M --> 200000 (30D40H)
;           T3B min = 20 mdtk
;           T3B max = 20000 mdtk
HITPO3: LCALL    HITV3A      ; panggil perhitungan kec. A posisi-3
      LCALL    HIT3AB      ; panggil perhitungan waktu AB posisi-3
      LCALL    HITV3B      ; panggil perhitungan kec. B posisi-3
      LCALL    HITAC3      ; panggil perhitungan percepatan posisi-3
      RET

; HITUNG KECEPATAN DI POSISI 3 TITIK A
;
HITV3A: MOV     OP3,#00H      ; Jarak 1 m
      MOV     OP2,#X32
      MOV     OP1,#X31
      MOV     OP0,#X30
      MOV     ANGKA1,MDT3A1   ; Waktu di titik A posisi-3
      MOV     ANGKA0,MDT3A0   ; MDT3A1 MDT3A0
      LCALL    DIV24          ; jarak / waktu
      MOV     V3A1,OP1        ; simpan hasil pembagian di
      MOV     V3A0,OP0        ; V3A1 V3A0 (16 bit)

```



```

; HITUNG KECEPATAN RATA-RATA DI TITIK 0 - A POSISI-3
;
MOV     OP3,#00
MOV     OP2,#00
MOV     ANGKA1,#00      ; pembagi = 2
MOV     ANGKA0,#02
LCALL   DIV24           ; (V3A1 V3A0) / 2
MOV     V3AR1,OP1       ; simpan hasil pembagian di
MOV     V3AR0,OP0       ; V3AR1 V3AR0
RET

;
; HITUNG T3AB
;
HIT3AB: MOV     OP2,#00      ; ambil waktu di titik B posisi-3
MOV     OP1,MDT3B1         ; MDT3B1 MDT3B0
MOV     OP0,MDT3B0
MOV     ANGKA2,#00
MOV     ANGKA1,MDT3A1      ; ambil waktu di titik A posisi-3
MOV     ANGKA0,MDT3A0      ; MDT3A1 MDT3A0
LCALL   SUBB24            ; (MDT3B1 MDT3B0) - (MDT3A1 MDT3A0)
MOV     T3AB1,OP1         ; simpan hasil pengurangan di
MOV     T3AB0,OP0         ; T3AB1 T3AB0 (16 bit)
RET

;
; HITUNG KECEPATAN DI POSISI 3 TITIK B
;
HITV3B: MOV     OP3,#00H    ; Jarak 1 m
MOV     OP2,#X32
MOV     OP1,#X31
MOV     OP0,#X30
MOV     ANGKA1,T3AB1      ; ambil waktu antara titik A-B posisi-3
MOV     ANGKA0,T3AB0      ; T3AB1 T3AB0
LCALL   DIV24            ; jarak / waktu
MOV     V3B1,OP1         ; simpan hasil pembagian di
MOV     V3B0,OP0         ; V3B1 V3B0 (16 bit)

;
; HITUNG KECEPATAN RATA-RATA DI TITIK A-B POSISI-3
;
MOV     OP2,#00
MOV     ANGKA2,#00      ; ambil kecepatan di titik A posisi-3
MOV     ANGKA1,V3A1      ; V3A1 V3A0
MOV     ANGKA0,V3A0
LCALL   ADD24           ; (V3AB1 V3AB0) + (V3A1 V3A0)

MOV     OP3,#00
MOV     ANGKA2,#00      ; pembagi = 2
MOV     ANGKA1,#00
MOV     ANGKA0,#02
LCALL   DIV24           ; [(V3AB1 V3AB0) + (V3A1 V3A0)]/2
MOV     V3BR1,OP1       ; simpan hasil pembagian di
MOV     V3BR0,OP0       ; V3BR1 V3BR0
RET

;
; HITUNG PERCEPATAN DI POSISI 3 TITIK B
;
HITAC3: NOP

;
; HITUNG V3A * V3A
;
MOV     OP2,#00
MOV     OP1,V3AR1
MOV     OP0,V3AR0
MOV     R3,V3AR1
MOV     R2,V3AR0
DEC     R2
LCALL   MULT16          ; (V3AR1 V3AR0) * (V3AR1 V3AR0)
MOV     BUF2,OP2        ; simpan hasil perkalian di
MOV     BUF1,OP1        ; BUF2 BUF1 BUF0 (24 bit)
MOV     BUF0,OP0

```

```

; HITUNG V3B * V3B
;
    MOV     OP2, #00
    MOV     OP1, V3BR1
    MOV     OP0, V3BR0
    MOV     R3, V3BR1
    MOV     R2, V3BR0
    DEC     R2
    LCALL   MULT16      ; (V3BR1 V3BR0) * (V3BR1 V3BR0)
    MOV     BUF7, OP2    ; simpan hasil perkalian di
    MOV     BUF6, OP1    ; BUF7 BUF6 BUF5 (24 bit)
    MOV     BUF5, OP0
;
; HITUNG V3B*V3B - V3A*V3A
;
    CLR     C
    MOV     KSIGN3, #20H
    MOV     A, V3B1
    MOV     B, V3A1
    CJNE    A, B, J3BSR1 ; cek apakah V3B1 = V3A1
    MOV     A, V3B0
    MOV     B, V3A0
    CJNE    A, B, J3BSR1 ; cek apakah V3B0 = V3A0
    MOV     ACCL32, #00  ; bila sama maka,
    MOV     ACCL31, #00  ; percepatan = 0
    MOV     ACCL30, #00
    RET

J3BSR1: JNC     J3BES1
;
; PERCEPATAN NEGATIF
; (V3AR1 V3AR0)^2 - (V3BR1 V3BR0)^2
;
    MOV     KSIGN3, #2DH ; TANDA '-'
    MOV     OP2, BUF2
    MOV     OP1, BUF1
    MOV     OP0, BUF0
    MOV     ANGKA2, BUF7
    MOV     ANGKA1, BUF6
    MOV     ANGKA0, BUF5
    LJMP    J3SB24

;
; PERCEPATAN POSITIF
; (V3BR1 V3BR0)^2 - (V3AR1 V3AR0)^2
;
J3BES1: MOV     KRSIGN, #20H ;
    MOV     OP2, BUF7
    MOV     OP1, BUF6
    MOV     OP0, BUF5
    MOV     ANGKA2, BUF2
    MOV     ANGKA1, BUF1
    MOV     ANGKA0, BUF0

J3SB24: LCALL   SUBB24
;
; (V3B^2-V3A^2) / (2*X)
; 2*X = 2*1 m = 2 m
; a = (V1B^2-V1A^2)/2 (4 DIGIT DI BELAKANG KOMA)
    MOV     ANGKA2, #00 ; PEMBAGI = 2
    MOV     ANGKA1, #00
    MOV     ANGKA0, #02
    LCALL   DIV24      ; (V1B^2-V1A^2)/2
    MOV     ACCL32, OP2 ; simpan hasil pembagian di
    MOV     ACCL31, OP1 ; ACCL32 ACCL31 ACCL30
    MOV     ACCL30, OP0
    RET

; *** TAMPILAN HASIL

```

```

;
TMPHSL: NOP
        LCALL  TPLP01      ; panggil tampilan posisi-1
        LCALL  TPLP02      ; panggil tampilan posisi-2
        LCALL  TPLP03      ; panggil tampilan posisi-3
        RET

;
; *** TAMPILAN DI POSISI 1
;
TPLP01: MOV    DPTR, #SPOS1A ; tampil posisi 1-A di baris-1
        MOV    BARIS, #01
        LCALL  TPLSTR

        MOV    DPTR, #STIM1A ; tampil waktu titik A posisi-1
        MOV    BARIS, #02    ; di baris-2
        LCALL  TPLSTR
        LCALL  TPTM1A
        LCALL  DELAY        ; aktifkan waktu tunda

        MOV    DPTR, #SVEL1A ; tampil kecepatan titik A posisi-1
        MOV    BARIS, #02    ; di baris-2
        LCALL  TPLSTR
        LCALL  TPLV1A
        LCALL  DELAY        ; aktifkan waktu tunda

;
; TAMPIL POSISI 1 TIAB
;
        MOV    DPTR, #SPOS1B ; tampil posisi 1-B di baris-1
        MOV    BARIS, #01
        LCALL  TPLSTR

        MOV    DPTR, #STM1AB ; tampil waktu titik B posisi-1
        MOV    BARIS, #02    ; di baris-2
        LCALL  TPLSTR
        LCALL  TPT1AB
        LCALL  DELAY        ; aktifkan waktu tunda

        MOV    DPTR, #SVEL1B ; tampil kecepatan titik B posisi-1
        MOV    BARIS, #02
        LCALL  TPLSTR
        LCALL  TPLV1B
        LCALL  DELAY        ; aktifkan waktu tunda

;
; TAMPIL POSISI 1 PERCEPATAN
;
        MOV    DPTR, #SPOS1C ; tampil percepatan di posisi-1
        MOV    BARIS, #01
        LCALL  TPLSTR
        MOV    DPTR, #SACCL1
        MOV    BARIS, #02
        LCALL  TPLSTR
        MOV    KRSIGN, KSIGN1
        LCALL  TPLAC1
        LCALL  DELAY        ; aktifkan waktu tunda
        RET

;
; tampilan waktu titik A posisi-1
;
TPTM1A: MOV    ANGKA2, #00
        MOV    ANGKA1, MDT1A1
        MOV    ANGKA0, MDT1A0
        LCALL  HEXDES
        LCALL  BCDKAR
        LCALL  TPLTIM
        RET

;
; tampilan waktu titik A-B posisi-1
;

```

```

TPTLAB: MOV     ANGKA2, #00
        MOV     ANGKA1, TLAB1
        MOV     ANGKA0, TLAB0
        LCALL   HEXDES
        LCALL   BCDKAR
        LCALL   TPLTIM
        RET

;
; tampilan kecepatan titik A posisi-1
;
TPLV1A: MOV     ANGKA2, #00
        MOV     ANGKA1, V1A1
        MOV     ANGKA0, V1A0
        LCALL   HEXDES
        LCALL   BCDKAR
        LCALL   TPLVEL
        RET

;
; tampilan kecepatan titik A-B posisi-1
;
TPLV1B: MOV     ANGKA2, #00
        MOV     ANGKA1, V1B1
        MOV     ANGKA0, V1B0
        LCALL   HEXDES
        LCALL   BCDKAR
        LCALL   TPLVEL
        RET

;
; tampilan percepatan A posisi-1
;
TPLAC1: MOV     ANGKA2, ACCL12
        MOV     ANGKA1, ACCL11
        MOV     ANGKA0, ACCL10
        LCALL   HEXDES
        LCALL   BTOKAC
        LCALL   TPLACC
        RET

;
; *** TAMPILAN DI POSISI 2
;
TPLPO2: MOV     DPTR, #SPOS2A ; tampil posisi 2-A di baris-1
        MOV     BARIS, #01
        LCALL   TPLSTR

        MOV     DPTR, #STIM2A ; tampil waktu titik A posisi-2
        MOV     BARIS, #02    ; di baris-2
        LCALL   TPLSTR
        LCALL   TPTM2A
        LCALL   DELAY         ; aktifkan waktu tunda

        MOV     DPTR, #SVEL2A ; tampil kecepatan titik A posisi-2
        MOV     BARIS, #02    ; di baris-2
        LCALL   TPLSTR
        LCALL   TPLV2A
        LCALL   DELAY         ; aktifkan waktu tunda

;
; TAMPIL POSISI 2. T2AB
;
        MOV     DPTR, #SPOS2B
        MOV     BARIS, #01
        LCALL   TPLSTR

        MOV     DPTR, #STM2AB
        MOV     BARIS, #02
        LCALL   TPLSTR

        LCALL   TPT2AB
        LCALL   DELAY

```

```

MOV     DPTR, #SVEL2B
MOV     BARIS, #02
LCALL   TPLSTR

LCALL   TPLV2B
LCALL   DELAY

;
; TAMPIL POSISI 2 PERCEPATAN
;
MOV     DPTR, #SPOS2C
MOV     BARIS, #01
LCALL   TPLSTR
MOV     DPTR, #SACCL2
MOV     BARIS, #02
LCALL   TPLSTR
MOV     KRSIGN, KRSIGN2
LCALL   TPLAC2
LCALL   DELAY
RET

;
; tampilan waktu titik A posisi-2
;
TPTM2A: MOV     ANGKA2, #00
MOV     ANGKA1, MDT2A1
MOV     ANGKA0, MDT2A0
LCALL   HEXDES
LCALL   BCDKAR
LCALL   TPLTIM
RET

;
; tampilan waktu titik A-B posisi-2
;
TPT2AB: MOV     ANGKA2, #00
MOV     ANGKA1, T2AB1
MOV     ANGKA0, T2AB0
LCALL   HEXDES
LCALL   BCDKAR
LCALL   TPLTIM
RET

;
; tampilan kecepatan titik A posisi-2
;
TPLV2A: MOV     ANGKA2, #00
MOV     ANGKA1, V2A1
MOV     ANGKA0, V2A0
LCALL   HEXDES
LCALL   BCDKAR
LCALL   TPLVEL
RET

;
; tampilan kecepatan titik B posisi-2
;
TPLV2B: MOV     ANGKA2, #00
MOV     ANGKA1, V2B1
MOV     ANGKA0, V2B0
LCALL   HEXDES
LCALL   BCDKAR
LCALL   TPLVEL
RET

;
; tampilan percepatan posisi-2
;
TPLAC2: MOV     ANGKA2, ACCL22
MOV     ANGKA1, ACCL21

```

```

MOV     ANGKA0, ACCL20
LCALL   HEXDES
LCALL   BTOKAC
LCALL   TPLACC
RET

;
; *** TAMPILAN DI POSISI 3
;
TLP03: MOV     DPTR, #SPOS3A
        MOV     BARIS, #01
        LCALL   TPLSTR

        MOV     DPTR, #STM3A
        MOV     BARIS, #02
        LCALL   TPLSTR

        LCALL   TPTM3A
        LCALL   DELAY

        MOV     DPTR, #SVEL3A
        MOV     BARIS, #02
        LCALL   TPLSTR

        LCALL   TPLV3A
        LCALL   DELAY

;
; TAMPIL POSISI 3 T3AB
;
        MOV     DPTR, #SPOS3B
        MOV     BARIS, #01
        LCALL   TPLSTR

        MOV     DPTR, #STM3AB
        MOV     BARIS, #02
        LCALL   TPLSTR

        LCALL   TPT3AB
        LCALL   DELAY

        MOV     DPTR, #SVEL3B
        MOV     BARIS, #02
        LCALL   TPLSTR

        LCALL   TPLV3B
        LCALL   DELAY

;
; TAMPIL POSISI 3 PERCEPATAN
;
        MOV     DPTR, #SPOS3C
        MOV     BARIS, #01
        LCALL   TPLSTR
        MOV     DPTR, #SACCL3
        MOV     BARIS, #02
        LCALL   TPLSTR
        MOV     KRSIGN, KSIGN3
        LCALL   TPLAC3
        LCALL   DELAY
        RET

;
; tampilan waktu titik A posisi-3
;
TPTM3A: MOV     ANGKA2, #00
        MOV     ANGKA1, MDT3A1
        MOV     ANGKA0, MDT3A0
        LCALL   HEXDES
        LCALL   BCDKAR

```

```

        LCALL  TPLTIM
        RET

;
; tampilan waktu titik A-B posisi-3
;
TPT3AB: MOV     ANGKA2, #00
        MOV     ANGKA1, T3AB1
        MOV     ANGKA0, T3AB0
        LCALL   HEXDES
        LCALL   BCDKAR
        LCALL   TPLTIM
        RET

;
; tampilan kecepatan titik A posisi-3
;
TPLV3A: MOV     ANGKA2, #00
        MOV     ANGKA1, V3A1
        MOV     ANGKA0, V3A0
        LCALL   HEXDES
        LCALL   BCDKAR
        LCALL   TPLVEL
        RET

;
; tampilan kecepatan titik B posisi-3
;
TPLV3B: MOV     ANGKA2, #00
        MOV     ANGKA1, V3B1
        MOV     ANGKA0, V3B0
        LCALL   HEXDES
        LCALL   BCDKAR
        LCALL   TPLVEL
        RET

;
; tampilan percepatan posisi-3
;
TPLAC3: MOV     ANGKA2, ACCL32
        MOV     ANGKA1, ACCL31
        MOV     ANGKA0, ACCL30
        LCALL   HEXDES
        LCALL   BTOKAC
        LCALL   TPLACC
        RET

; *****
; KONVERSI HEKSA KE DESIMAL
; BIL HEXA : ANGKA2 ANGKA1 ANGKA0
; DIUBAH KE RATRIB PULRIB RIBUAN RATUS PULUH SATUAN
;
HEXDES: MOV     R7, #00H
        MOV     R6, #00H
        MOV     R5, #00H

        MOV     SATUAN, #00
        MOV     PULUH, #00
        MOV     RATUS, #00
        MOV     RIBUAN, #00
        MOV     PULRIB, #00
        MOV     RATRIB, #00

ULANGA: MOV     A, R7
        MOV     B, ANGKA2
        CJNE    A, B, JPINC1
        MOV     A, R6
        MOV     B, ANGKA1
        CJNE    A, B, JPINC1
        MOV     A, R5

```



```

MOV     B,ANGKAO
CJNE    A,B,JPINC1
RET

JPINC1: INC     R5
MOV     A,R5
MOV     B,#00H
CJNE    A,B,JNEXTA
INC     R6
MOV     A,R6
MOV     B,#00H
CJNE    A,B,JNEXTA
INC     R7
MOV     A,R7

JNEXTA: INC     SATUAN
MOV     A,SATUAN
MOV     B,#10
CJNE    A,B,ULANGA
MOV     SATUAN,#00
INC     PULUH
MOV     A,PULUH
MOV     B,#10
CJNE    A,B,ULANGA
MOV     PULUH,#00
INC     RATUS
MOV     A,RATUS
MOV     B,#10
CJNE    A,B,ULANGA
MOV     RATUS,#00
INC     RIBUAN
MOV     A,RIBUAN
MOV     B,#10
CJNE    A,B,ULANGA
MOV     RIBUAN,#00
INC     PULRIB
MOV     A,PULRIB
MOV     B,#10
CJNE    A,B,ULANGA
MOV     PULRIB,#00
INC     RATRIB
LJMP    ULANGA

```

```

; *****
;
; KONVERSI BCD 0-9 KE KARAKTER
;
BCDKAR: MOV     A,RATRIB
MOV     B,#00
CJNE    A,B,GONOR1
MOV     KAR1,#20H

MOV     A,PULRIB
MOV     B,#00
CJNE    A,B,GONOR2
MOV     KAR2,#20H

MOV     A,RIBUAN
MOV     B,#00
CJNE    A,B,GONOR3
MOV     KAR3,#20H
LJMP    GONOR4

GONOR1: ORL     A,#30H
MOV     KAR1,A

MOV     A,PULRIB
GONOR2: ORL     A,#30H
MOV     KAR2,A

```

```

      MOV      A,RIBUAN
GONOR3: ORL      A,#30H
      MOV      KAR3,A

      MOV      A,RATUS
      ORL      A,#30H
      MOV      KAR4,A

      MOV      A,PULUH
      ORL      A,#30H
      MOV      KAR5,A

      MOV      A,SATUAN
      ORL      A,#30H
      MOV      KAR6,A
      RET

; *****
;
; KONVERSI BCD 0-9 KE KARAKTER ACC
;
BTOKAC: MOV      A,RATRIB
      MOV      B,#00
      CJNE     A,B,GONOA1
      MOV      KAR1,#20H
      LJMP     GONOA2

GONOA1: ORL      A,#30H
      MOV      KAR1,A

GONOA2: MOV      A,PULRIB
      ORL      A,#30H
      MOV      KAR2,A

      MOV      A,RIBUAN
      ORL      A,#30H
      MOV      KAR3,A

      MOV      A,RATUS
      ORL      A,#30H
      MOV      KAR4,A

      MOV      A,PULUH
      ORL      A,#30H
      MOV      KAR5,A

      MOV      A,SATUAN
      ORL      A,#30H
      MOV      KAR6,A
      RET

; *****
;
; TAMPILAN FORMAT WAKTU
;
TPLTIM: MOV      POSX,#06
      LCALL     KEBR2X
      MOV      A,KAR1
      LCALL     TPLKAR
      MOV      A,KAR2
      LCALL     TPLKAR
      MOV      A,KAR3
      LCALL     TPLKAR
      MOV      A,KAR4
      LCALL     TPLKAR
      MOV      A,KAR5
      LCALL     TPLKAR
      MOV      A,KAR6
      LCALL     TPLKAR
      RET

```



```
; *****
; TAMPILAN FORMAT KECEPATAN
;
```

```
TPLVEL: MOV    POSX, #06
        LCALL  KEBR2X
        MOV    A, KAR2
        LCALL  TPLKAR
        MOV    A, KAR3
        LCALL  TPLKAR
        MOV    A, KAR4
        LCALL  TPLKAR
        MOV    A, #2EH
        LCALL  TPLKAR
        MOV    A, KAR5
        LCALL  TPLKAR
        MOV    A, KAR6
        LCALL  TPLKAR
        RET
```

```
; *****
;
; TAMPILAN FORMAT PERCEPATAN
;
```

```
TPLACC: MOV    POSX, #03
        LCALL  KEBR2X
        MOV    A, KRSIGN
        LCALL  TPLKAR
        MOV    A, KAR1
        LCALL  TPLKAR
        MOV    A, KAR2
        LCALL  TPLKAR
```

```
        MOV    A, #2EH
        LCALL  TPLKAR
```

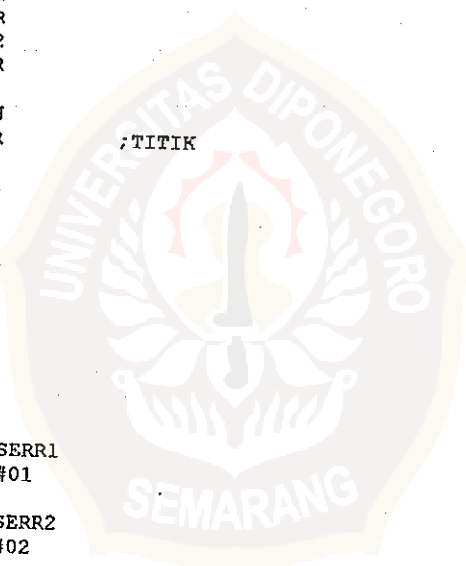
```
        MOV    A, KAR3
        LCALL  TPLKAR
        MOV    A, KAR4
        LCALL  TPLKAR
        MOV    A, KAR5
        LCALL  TPLKAR
        MOV    A, KAR6
        LCALL  TPLKAR
        RET
```

```
TMPERR: MOV    DPTR, #SERR1
        MOV    BARIS, #01
        LCALL  TPLSTR
        MOV    DPTR, #SERR2
        MOV    BARIS, #02
        LCALL  TPLSTR
        RET
```

```
MUL100: MOV    R3, #00
        MOV    R2, #99
        LCALL  MULT16
        RET
```

```
; *****
;
; INPUT    OP2 OP1 OP0
; OUTPUT (OP2 OP1 OP0) DIKALIKAN 1000
;
```

```
X1000: MOV    R3, #27H      ;9999 : 270FH
        MOV    R2, #0FH
        LCALL  MULT16
```



RET

```
;
; PENAMBAHAN OP2 OP1 OP0 DENGAN NILAI AWALNYA
; (DISIMPAN DI ANGKA2 ANGKA1 ANGKA0)
; SEBANYAK NILAI DI R3 (MSB) DAN R2 (LSB)
;
```

```
MULT16: MOV     ANGKA2, OP2
          MOV     ANGKA1, OP1
          MOV     ANGKA0, OP0
```

```
          MOV     DPTR, #00
ULADD1:  LCALL   ADD24
          INC     DPTR
          MOV     A, DPH
          MOV     B, R3
          CJNE    A, B, ULADD1
          MOV     A, DPL
          MOV     B, R2
          CJNE    A, B, ULADD1
          RET
```

```
; *****
```

```
; PENJUMLAHAN 24 BIT
```

```
ADD24:   CLR     C
          MOV     A, OP0
          MOV     B, ANGKA0
          ADDC    A, B
          MOV     OP0, A

          MOV     A, OP1
          MOV     B, ANGKA1
          ADDC    A, B
          MOV     OP1, A

          MOV     A, OP2
          MOV     B, ANGKA2
          ADDC    A, B
          MOV     OP2, A
          RET
```

```
; *****
```

```
; PEMBAGIAN 24 BIT
```

```
; YANG DIBAGI : OP3      OP2      OP1      OP0
; PEMBAGI      :          ANGKA1 ANGKA0
```

```
; HASIL BAGI  : OP3      OP2      OP1      OP0
; SISA        :          SISA1 SISA0
```

```
DIV24:   MOV     SISA1, #00
          MOV     SISA0, #00
          MOV     TMP3, #00
          MOV     TMP2, #00
          MOV     TMP1, #00
          MOV     TMP0, #00
          MOV     R2, #32      (* INDEKS *)
DIVLOP:  LCALL   SHIFTD
          MOV     A, SISA0
          RLC     A
          MOV     SISA0, A
          MOV     A, SISA1
          RLC     A
          MOV     SISA1, A
```

```
; TEST TO SEE IF SISA1:SISA0 >= ANGKA1:ANGKA0
          JC      KURANG
```

```

CLR      C
MOV      A,SISA1
MOV      B,ANGKA1
SUBB     A,B
JC       TETAP

; SISA1 > ANGKA1 ATAU SISA1 = ANGKA1
JNZ      KURANG

; IF SISA1 = ANGKA1, TEST FOR SISA0 >= ANGKA0
CLR      C
MOV      A,SISA0
MOV      B,ANGKA0
SUBB     A,B
JC       TETAP

KURANG: CLR      C
MOV      A,SISA0
MOV      B,ANGKA0
SUBB     A,B
MOV      SISA0,A
MOV      A,SISA1
MOV      B,ANGKA1
SUBB     A,B
MOV      SISA1,A
SETB     C
LJMP     QUOTIE

TETAP: CLR      C

QUOTIE: LCALL  SHIFQ
; TEST FOR COMPLETION
DJNZ     R2,DIVLOP

; NOW WE ARE ALL DONE
MOV      OP0,TMP0
MOV      OP1,TMP1
MOV      OP2,TMP2
MOV      OP3,TMP3
RET

;DIVIDENT
SHIFTD: CLR      C
MOV      A,OP0
RLC      A
MOV      OP0,A
MOV      A,OP1
RLC      A
MOV      OP1,A
MOV      A,OP2
RLC      A
MOV      OP2,A
MOV      A,OP3
RLC      A
MOV      OP3,A
RET

;QUOTIENT
SHIFQ: MOV      A,TMP0
RLC      A
MOV      TMP0,A
MOV      A,TMP1
RLC      A
MOV      TMP1,A
MOV      A,TMP2
RLC      A
MOV      TMP2,A
MOV      A,TMP3
RLC      A
MOV      TMP3,A
RET

```



```

; *****
;
; PENGURANGAN 24 BIT
;
SUBB24: CLR      C
        MOV      A, OP0
        MOV      B, ANGKA0
        SUBB     A, B
        MOV      OP0, A
        MOV      A, OP1
        MOV      B, ANGKA1
        SUBB     A, B
        MOV      OP1, A
        MOV      A, OP2
        MOV      B, ANGKA2
        SUBB     A, B
        MOV      OP2, A
        RET

; *****
;
; PEMBACAAN WAKTU
;
; HASIL : MILI DETIK MSB --> A
;         MILI DETIK LSB --> B
BCWKTU: MOV      A, TMDTK1
        MOV      B, TMDTK0
        MOV      R7, A
        MOV      R6, B
        MOV      A, TMDTK1
        MOV      B, TMDTK0
        MOV      R5, A
        MOV      R4, B

        MOV      A, R7
        MOV      B, R5
        CJNE     A, B, CKMDT0
        MOV      A, R7
        MOV      B, R6
        RET

CKMDT0: MOV      A, R5
        MOV      B, R4
        RET

; *****
;
; INTERRUPT INTO
;
;
RSTAWL: MOV      FLGAWL, #01
        RETI

; *****
;
; INTERRUPT TIMER-0
;
TIMER0: PUSH     ACC
        PUSH     B
        PUSH     DPL
        PUSH     DPH
        MOV      A, #00
        MOV      ACC.0, C
        MOV      BUFFRC, A

        INC      CLOCKS0
        MOV      A, CLOCKS0
        MOV      B, #04
        CJNE     A, B, JPQUIT

```

```

MOV     CLOCK0, #00      ; 1 milidetik
MOV     DPH, TMDTK1
MOV     DPL, TMDTK0
INC     DPTR
MOV     TMDTK1, DPH
MOV     TMDTK0, DPL
MOV     A, DPH
MOV     B, #4EH
CJNE    A, B, JPQUIT      ; CEK 60000 DES (4E20H)
MOV     A, DPL
MOV     B, #20H
CJNE    A, B, JPQUIT
MOV     FLAGERR, #01

JPQUIT: MOV     A, BUFERRC
MOV     C, ACC.0
POP     DPH
POP     DPL
POP     B
POP     ACC
RETI

; *****
;
; TAMPILAN LCD
;
; P0 : DATA
; P2.0 : ENABLE
; P2.1 : R/W
; P2.2 : RS
;
INISIA: MOV     P0, #38H
        LCALL    ENABLO
        MOV     P0, #0EH
        LCALL    ENABLO
        MOV     P0, #06H
        LCALL    ENABLO
        RET

DSPCLR: MOV     P0, #01H
        LCALL    ENABLO
        MOV     P0, #01H
        LCALL    ENABLO
        RET

CURSON: MOV     P0, #0EH
        LCALL    ENABLO
        RET

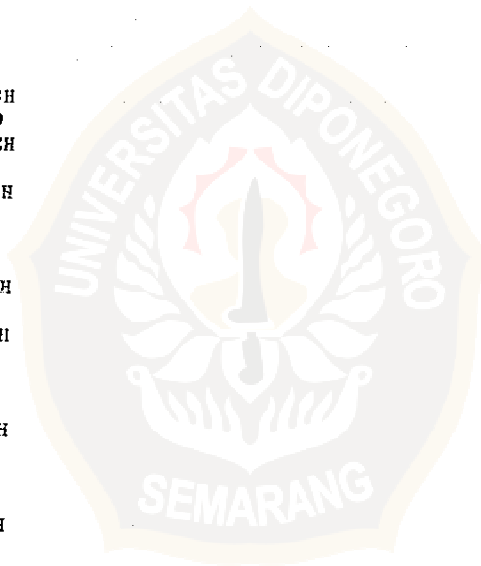
CURSOF: MOV     P0, #0CH
        LCALL    ENABLO
        RET

TMPAWL: MOV     DPTR, #SAWAL1
        MOV     BARIS, #01
        LCALL    TPLSTR
        MOV     DPTR, #SAWAL2
        MOV     BARIS, #02
        LCALL    TPLSTR
        RET

TPLSTR: MOV     POSX, #00
        MOV     A, BARIS
        MOV     B, #01
        CJNE    A, B, JPBR2
        LCALL    KEBR1X
        AJMP     NEXTTP

JPBR2:  LCALL    KEBR2X
NEXTTP: MOV     INDEX, #00

```



```

LTPSTR: MOV    A, INDEX
        MOVC   A, @A+DPTR
        LCALL  TPLKAR
        INC    INDEX
        MOV    A, INDEX
        MOV    B, #16
        CJNE   A, B, LTPSTR
        RET

; POSX : POSISI 0 - F
KEBR1X: MOV    A, POSX
        ORL    A, #80H
        MOV    P0, A
        LCALL  ENABLO
        RET

; POSX : POSISI 0 - F
KEBR2X: MOV    A, POSX
        ORL    A, #0C0H
        MOV    P0, A
        LCALL  ENABLO
        RET

TPLKAR: MOV    P0, A
        LCALL  ENABL1
        RET

ENABLO: CLR    P2.0          ; ENABLE
        CLR    P2.1          ; R/W
        CLR    P2.2          ; RS : 0
        LCALL  TDEL1
        SETB   P2.0
        LCALL  TDEL1
        CLR    P2.0
        RET

ENABL1: CLR    P2.0          ; ENABLE
        CLR    P2.1          ; R/W
        SETB   P2.2          ; RS : 1
        LCALL  TDEL1
        SETB   P2.0
        LCALL  TDEL1
        CLR    P2.0
        RET

; *****
;
; DELAY
;
TDEL1: MOV    R2, #2
LPDEL1: MOV    R3, #100
        DJNZ   R3, $
        DJNZ   R2, LPDEL1
        RET

DELO1: MOV    R2, #2
JDELO1: MOV    R3, #250
JDELO2: MOV    R4, #250
        DJNZ   R4, $
        DJNZ   R3, JDELO2
        DJNZ   R2, JDELO1
        RET

DELAY: MOV    R2, #24
JDEL1: MOV    R3, #250
JDEL2: MOV    R4, #250
        DJNZ   R4, $
        DJNZ   R3, JDEL2
        DJNZ   R2, JDEL1
        RET

```




```

; *****
;
; STRING TAMPILAN
;
;
SAWAL1: DB      ' JALANKAN MOBIL '
SAWAL2: DB      ' MULAI PENGUKURAN '
; STRING TAMPILAN POSISI 1
SPOS1A: DB      'X1A : 0 - 0.5 m'
SPOS1B: DB      'X1AB: 0.5 - 1 m'
SPOS1C: DB      'X1  : 0.5 - 1 m'
STM1A: DB       'T1A :          ms'
STM1AB: DB      'T1AB:          ms'
SVEL1A: DB      'V1A :          m/s'
SVEL1B: DB      'V1B :          m/s'
SACCL1: DB      'a1:          m/s^2'

; STRING TAMPILAN POSISI 2
SPOS2A: DB      'X2A : 0 - 0.75 m'
SPOS2B: DB      'X2AB: 0.75-1.5 m'
SPOS2C: DB      'X2  : 0.75-1.5 m'
STM2A: DB       'T2A:          ms'
STM2AB: DB      'T2AB:          ms'
SVEL2A: DB      'V2A:          m/s'
SVEL2B: DB      'V2B:          m/s'
SACCL2: DB      'a2:          m/s^2'

; STRING TAMPILAN POSISI 3
SPOS3A: DB      'X3A : 0 - 1 m'
SPOS3B: DB      'X3AB: 1 - 2 m'
SPOS3C: DB      'X3  : 0 - 1 - 2 m'

3A: DB 'T3A:          ms'
STM3AB: DB      'T3AB:          ms'
SVEL3A: DB      'V3A:          m/s'
SVEL3B: DB      'V3B:          m/s'
SACCL3: DB      'a3:          m/s^2'

; STRING TAMPILAN ERROR
SERR1: DB      'ERROR.....'
SERR2: DB      'ULANGI DARI AWAL'

```

