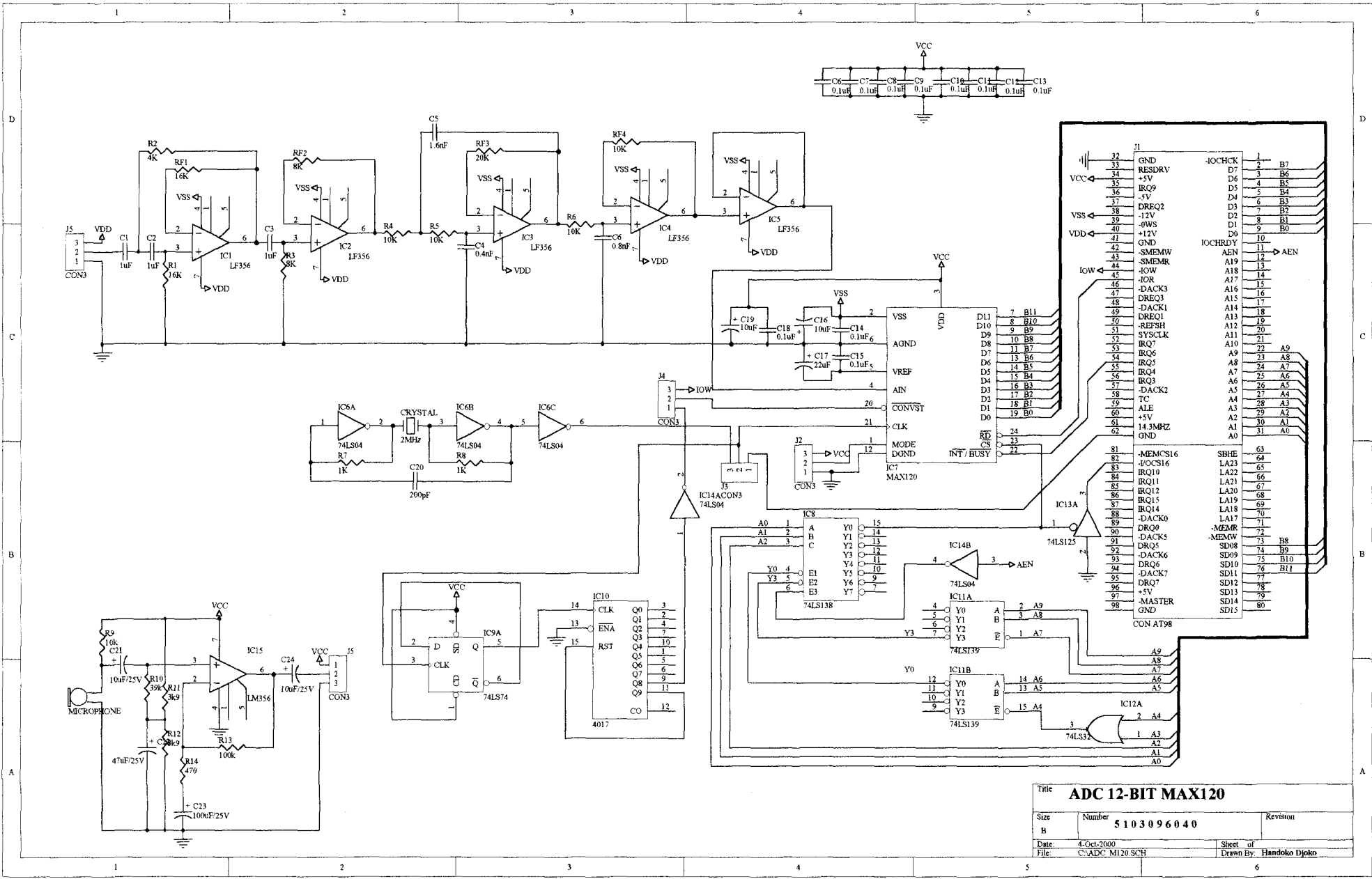
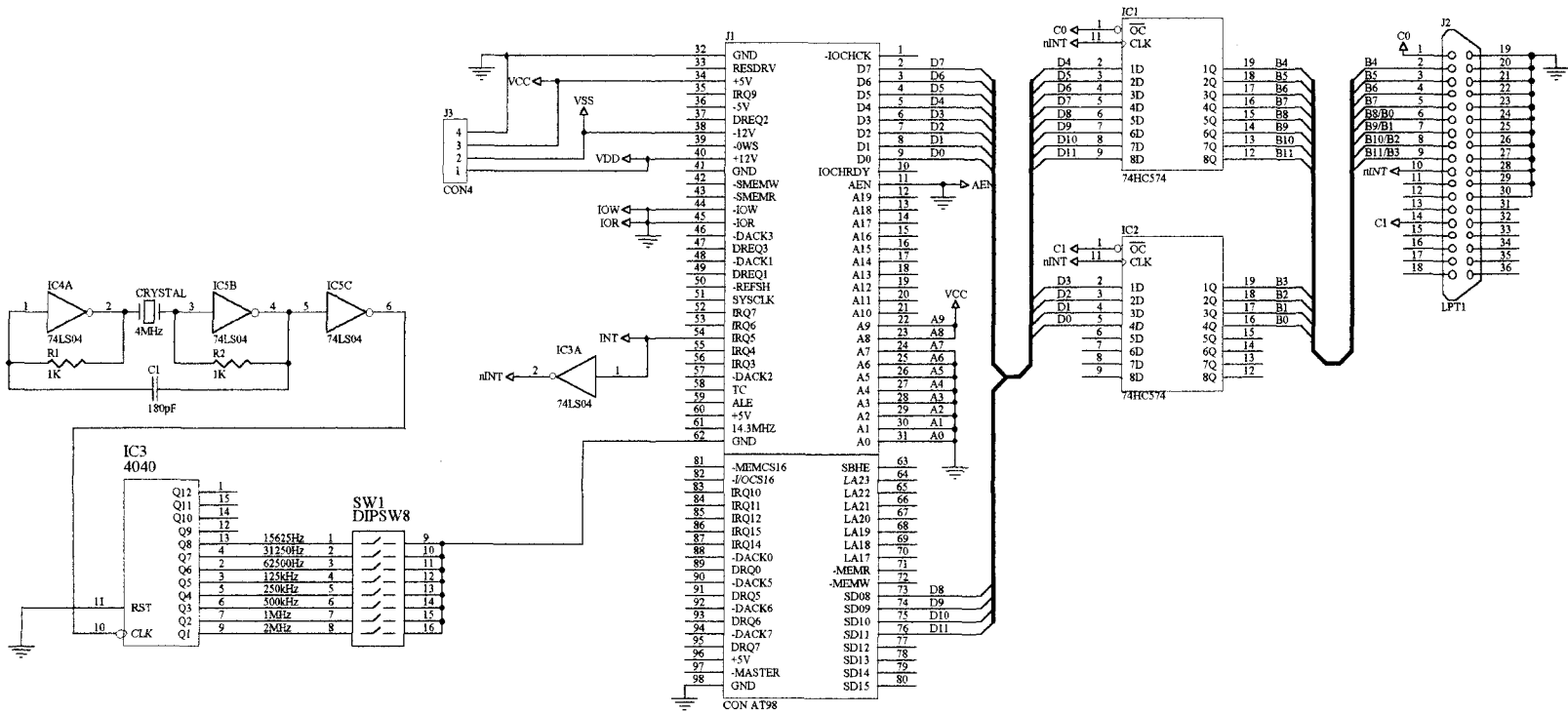


LAMPIRAN

LAMPIRAN A



Title			ADC 12-BIT MAX120		
Size	Number	Revision			
B	5103096040				
Date:	4-Oct-2000	Sheet of			
File:	C:\ADC MI20 SCH	Drawn By:		Handoko Djoko	



Title		
PARALEL PRINTER INTERFACE		
Size	Number	Revision
B	5103096040	
Date:	4-Oct-2000	Sheet of
File:	C:\PARALEL SCH	Drawn By: Handoko Djoko

LAMPIRAN C

```
{*****}
Program FFT Radix-2 Decimation in Frequency (DIF)
Author   :   Handoko Djoko
          :   5103096040
Abstrak  :   Menyempel data digital dengan menggunakan ADC dan
            :   memproses data tersebut dengan menggunakan metoda
            :   logatitmik untuk menghitung gain dan metoda FFT untuk
            :   mengetahui spektrum sinyal pada domain frekuensi untuk
            :   frekuensi audio (20Hz - 20kHz) serta menampilkan pada
            :   layar monitor PC dengan tampilan grafis.
*****}

uses crt, dos, graph;

type complex = record
    re      : real;
    im      : real;
end;

type str8 = string[8];

const data      = $378;      {port alamat data paralel printer}
      status    = $379;      {port alamat status paralel printer}
      control   = $37A;      {port alamat kontrol paralel ptinter}
      ecr       = data+$402; {port alamat Extended Control Register
printer}
      sampel    = 1024;      {jumlah titik sampel}
      m         = 10;        {jumlah bit dari titik sampel}
      Gray50    : FillPatternType =
($AA,$55,$AA,$55,$AA,$55,$AA,$55);

var GrDriver, GrMode, ErrCode : integer;
    n, p, sampling, max, min  : integer;
    save                      : pointer;
    tombol                    : char;
    datastr                   : str8;
    a                         : array [0..sampel] of complex;
    data1, data2              : array [0..sampel] of byte;
    data3, data_max, data_min : array [0..sampel] of longint;
    data_total                : array [0..sampel] of longint;
    Vpp, Vpp2, dB, dB2, gain : array [0..sampel] of real;

procedure data_in;{prosedur untuk input data}
begin
    port[ecr] := port[ecr] or $20;
    port[control] := port[control] or $20;
end;

procedure data_out;{prosedur untuk output data}
begin
    port[ecr] := port[ecr] and $DF;
    port[control] := port[control] or $DF;
end;

procedure aktif_IRQ7;{prosedur untuk enable interrupt}
begin
```

```

    port[$21] := port[$21] and $7F;
    port[control] := port[control] or $10;
end;

procedure matikan_IRQ7;{prosedur untuk disable interrupt}
begin
    port[$21] := port[$21] or $80;
    port[control] := port[control] and $EF;
end;

procedure int_kembali;{prosedure untuk kembali ke interupsi awal}
begin
    matikan_IRQ7;
    SetIntVec($0F,save);
end;

procedure data_high;{prosedure untuk mengambil data high (D4 - D11)}
begin
    port[control] := port[control] and $FE;
    port[control] := port[control] or $21;
end;

procedure data_low;{prosedure untuk mengambil data low (D0 - D3)}
begin
    port[control] := port[control] and $FD;
    port[control] := port[control] or $22;
end;

procedure baca_adc;{prosedur untuk men-konversi data digital
                    dari ADC. Interrupt yang digunakan yaitu IRQ7
                    yang meng-interupsi setiap 1024 titik sampel
                    yang diambil}

interrupt;
begin
    data_in;
    data_high;
    data1[n] := port[data] and $0FF;
    data_low;
    data2[n] := port[data] and $00F;
    data3[n] := data1[n] shl 4;
    data3[n] := data3[n] or data2[n];
    data3[n] := data3[n] and $FFF;
    if data3[n] and $800 = $800 then
        data3[n] := -2048+(data3[n] and $7FF);
    inc(n);
    if n >= sampel then
        begin
            n := 0;
            sampling := 1;
            matikan_IRQ7;           {disable IRQ7}
        end;
        port[$20] := $20;         {end of interrupt}
    end;
end;

procedure int_save;{prosedur untuk menyimpan alamat IRQ7 ke save
dan

```

```

                                mengeset alamat IRQ7}
begin
  GetIntVec($0F,save);
  SetIntVec($0F,@baca_adc);
end;

procedure Inisialisasi_graph;{prosedur mengaktifkan mode grafik
                                pada bahasa pemrograman Pascal
                                dan mode resolusi yang digunakan
yaitu 640 x 480 pixel}
begin
  GrDriver := Detect;
  InitGraph(GrDriver, GrMode, 'd:');
  ErrCode := GraphResult;
  if GraphResult <> Gok then Halt;
end;

procedure mul(var z : complex; x, y : complex);
    {prosedur menghitung penjumlahan dan perkalian bilangan
    kompleks}
begin
  z.re := x.re*y.re - x.im*y.im;
  z.im := x.re*y.im + x.im*y.re;
end;

procedure sub(var z : complex; x, y : complex);
    {prosedur menghitung pengurangan bilangan kompleks}
begin
  z.re := x.re - y.re;
  z.im := x.im - y.im;
end;

procedure add(var z : complex; x, y : complex);
    {prosedur menghitung penjumlahan bilangan kompleks}
begin
  z.re := x.re + y.re;
  z.im := x.im + y.im;
end;

function abs(x : complex) : real;
    {prosedur menghitung nilai total atau magnitude bilangan
    kompleks}
begin
  abs := sqrt(sqr(x.re) + sqr(x.im))
end;

procedure bit_reverse;{prosedur mengubah urutan bilangan kompleks}
var i, j, k : word;
    t      : complex;
begin
  j := 1;
  for i := 1 to sampel - 1 do
  begin
    if i < j then
    begin
      t := a[j];

```

```

        a[j] := a[i];
        a[i] := t;
    end;
    k := sampel shr 1;
    while k < j do
    begin
        j := j - k;
        k := k shr 1;
    end;
    j := j + k;
end;
end;

```

```

procedure fft;{prosedur mengkonversi bilangan kompleks dari domain
                waktu ke domain frekuensi. Disini bilangan kompleks
                dikalikan dan dijumlahkan dengan faktor dari
                bilangan kompleks dan diproses pengurangan dan
                penjumlahan bilangan kompleks}

```

```

var u, w, t : complex;
    i, j, k, le, le1, p : word;
begin
    for k := 1 to m do
    begin
        le := 1 shl k;
        le1 := le shr 1;
        u.re := 1; u.im := 0;
        w.re := cos(pi/le1);
        w.im := -sin(pi/le1);
        for j := 1 to le1 do
        begin
            i := j;
            repeat
                p := i + le1;
                mul(t, a[p], u);
                sub(a[p], a[i], t);
                add(a[i], a[i], t);
                inc(i, le);
            until i >= sampel;
            mul(u, u, w);
        end;
    end;
end;
end;

```

```

procedure area;{prosedur untuk membuat area untuk sinyal yang akan
                ditampilkan ke layar}

```

```

begin
    SetFillPattern(Gray50,1);
    Bar(50,3,630,57);
    SetFillPattern(Gray50,8);
    Bar(50,60,630,360);
    SetLineStyle(DottedLn,0,NormWidth);
    setcolor(8);
    for p := 0 to 19 do
        line(50+(p*29),60,50+(p*29),360);
        line(50,60,630,60);line(50,94,630,94);line(50,125,630,125);
        line(50,151,630,151);line(50,174,630,174);line(50,196,630,196);
    end;
end;

```

```

line(50,214,630,214);line(50,230,630,230);line(50,245,630,245);
line(50,259,630,259);line(50,271,630,271);line(50,281,630,281);
line(50,289,630,289);line(50,296,630,296);line(50,325,630,325);
line(50,341,630,341);
setcolor(3);
line(50,60,630,60);line(50,66,630,66);line(50,76,630,76);
line(50,86,630,86);line(50,96,630,96);line(50,106,630,106);
line(50,116,630,116);line(50,126,630,126);line(50,136,630,136);
line(50,146,630,146);line(50,156,630,156);line(50,166,630,166);
line(50,176,630,176);line(50,186,630,186);line(50,196,630,196);
line(50,246,630,246);line(50,296,630,296);line(50,346,630,346);
SetLineStyle(SolidLn,0,NormWidth);
setcolor(8);
line(50,60,630,60);
line(630,60,630,360);
setcolor(4); line(50,60,50,360); {garis vertikal}
line(48,360,630,360); {garis horizontal bawah}
setcolor(8);
line(48,60,52,60);line(48,94,52,94);line(48,125,52,125);
line(48,151,52,151);line(48,174,52,174);line(48,196,52,196);
line(48,214,52,214);line(48,230,52,230);line(48,245,52,245);
line(48,259,52,259);line(48,271,52,271);line(48,281,52,281);
line(48,289,52,289);line(48,296,52,296);line(48,325,52,325);
line(48,341,52,341);
setcolor(11);
line(48,66,52,66);line(48,76,52,76);line(48,86,52,86);
line(48,96,52,96);line(48,106,52,106);line(48,116,52,116);
line(48,126,52,126);line(48,136,52,136);line(48,146,52,146);
line(48,156,52,156);line(48,166,52,166);line(48,176,52,176);
line(48,186,52,186);line(48,196,52,196);line(48,246,52,246);
line(48,296,52,296);line(48,346,52,346);
setcolor(7); rectangle(50,3,630,57);
setcolor(8); line(50,30,630,30);
setcolor(11);
line(628,5,632,5);line(628,30,632,30);line(628,55,632,55);
line(48,5,52,5);line(48,30,52,30);line(48,55,52,55);
setcolor(7);
outtextxy(0,4,' 5V');
outtextxy(0,27,' 0V');
outtextxy(0,50,' -5V');
setcolor(7);
outtextxy(0,170,' G');
outtextxy(0,180,' a');
outtextxy(0,190,' i');
outtextxy(0,200,' n');
outtextxy(0,218,' (dB) ');
setcolor(8);
outtextxy(0,58,' 13');
outtextxy(0,91,' 12');
outtextxy(0,122,' 11');
outtextxy(0,148,' 10');
outtextxy(0,171,' 9');
outtextxy(0,193,' 8');
outtextxy(0,211,' 7');
outtextxy(0,227,' 6');
outtextxy(0,242,' 5');

```



```

    outtextxy(0,256,' 4');
    outtextxy(0,268,' 3');
    outtextxy(0,278,' 2');
    outtextxy(0,286,' 1');
    outtextxy(0,294,' 0');
    outtextxy(0,322,' -5');
    outtextxy(0,338,' -10');
    outtextxy(0,357,' -20');
setcolor(11);
    outtextxy(2,63,' 13');
    outtextxy(2,93,' 10');
    outtextxy(2,143,' 5');
    outtextxy(2,193,' 0');
    outtextxy(2,243,' -5');
    outtextxy(2,293,' -10');
    outtextxy(2,343,' -15');
setcolor(8);
    for p := 0 to 20 do
        line(50+(p*29),362,50+(p*29),358);
setcolor(8);
    outtextxy(39,370,' 0');
    outtextxy(68,370,' 1');
    outtextxy(97,370,' 2');
    outtextxy(126,370,' 3');
    outtextxy(154,370,' 4');
    outtextxy(184,370,' 5');
    outtextxy(213,370,' 6');
    outtextxy(242,370,' 7');
    outtextxy(271,370,' 8');
    outtextxy(300,370,' 9');
    outtextxy(333,370,'10');
    outtextxy(362,370,'11');
    outtextxy(391,370,'12');
    outtextxy(420,370,'13');
    outtextxy(449,370,'14');
    outtextxy(478,370,'15');
    outtextxy(507,370,'16');
    outtextxy(536,370,'17');
    outtextxy(565,370,'18');
    outtextxy(594,370,'19');
    outtextxy(620,370,'20');
setcolor(7);
outtextxy(270,384,'Frekuensi (kHz)');
setcolor(10);
setcolor(4); rectangle(482,382,632,408);
setcolor(15); outtextxy(486,386,'Vpp =          V ');
                outtextxy(486,398,'Gain =          dB');
setcolor(11); outtextxy(48,400,'< Tekan "Enter" untuk HOLD/RUN >');
SetLineStyle(DottedLn,0,NormWidth);
end;

procedure rumus_gain_high; Near;
begin
    if data3[p] > data_max[max] then
        begin
            data_max[max] := data3[p];

```

```

end;
Vpp[p] := (data_max[max]+0.00001)/204.8;
dB[p] := 20*(ln((Vpp[p])/(2*sqrt(2)))/0.775)/ln(10));
port[$20]:=$20;
end;

procedure rumus_gain_low; Near;
begin
  if data3[p] < data_min[min] then
  begin
    data_min[min] := data3[p];
  end;
  Vpp2[p] := ((-1)*data_min[min]+0.00001)/204.8;
  dB2[p] := 20*(ln((Vpp2[p])/(2*sqrt(2)))/0.775)/ln(10));
  port[$20]:=$20;
end;

procedure gain_dB; Near;
begin
  data_total[p] := (data_max[max]-data_min[min]);
  Vpp2[p] := (Vpp[p]+Vpp2[p])/2;
  gain[p] := (dB[p]+dB2[p])/2;
  if gain[p] <= -16.3943 then gain[p] := -16.3943;
end;

procedure tulis_gain; Near;
begin
  str(Vpp2[p]:1:4,datastr);
  outtextxy(546,386,datastr);
  str(gain[p]:4:4,datastr);
  outtextxy(546,398,datastr);
end;

begin
  clrscr;
  tombol := #0;
  textcolor(11);
  gotoxy(22,12);writeln('Tekan "Enter" untuk mulai pengukuran');
  repeat
    if keypressed then tombol := readkey;
  until (tombol = #13);
  if tombol = #13 then tombol := #0;
  Inisialisasi_graph;
  setcolor(4); rectangle(1,1,639,479);
  rectangle(10,10,220,28);
  rectangle(510,45,632,58);
  rectangle(466,6,626,30);
  setcolor(11); outtextxy(20,15,'* Grafik Gain dalam dB *');
  setcolor(10); outtextxy(516,48,'~Sinyal Input~');
  setcolor(10); outtextxy(470,10,' -');
  setcolor(7); outtextxy(470,10,'( ) Grafik Gain');
  setcolor(14); outtextxy(470,20,' -');
  setcolor(7); outtextxy(470,20,'( ) Grafik Spektrum');
  SetViewPort(2,60,638,468,true);
  area;
  n := 0;

```

```

p := 0;
sampling := 0;
int_save;          {simpan alamat IRQ7 ke save}
aktif_IRQ7;       {enable IRQ7}
port[data] := 0;  {keadaan awal data}
repeat
until sampling = 1;
for p := 0 to sampel do {mengubah nilai bit ADC ke bilangan
                        kompleks FFT}
begin
  a[p].re := ((10*data3[p])/1688)*(1-cos(2*pi*(p shl 1)/sampel));
  a[p].im := 0;
end;
bit_reverse;
fft;
p := 0;
repeat
  rumus_gain_high;
  rumus_gain_low;
  gain_dB;
  inc(p);
  if sampling = 1 then {menampilkan data ke layar monitor PC}
  begin
    putpixel(50+p,196-round(10*gain[p]),10);
    setcolor(10);
    moveto(50+p,30+(data3[p-1]) div 80);
    lineto(50+p,30+(data3[p]) div 80);
    if p < 579 shl 1 then {menampilkan data yang telah diubah ke
                          bilangan kompleks FFT}
    begin
      setcolor(14);
      moveto(50+p,360-round(50*abs(a[p-1])/sampel));
      lineto(50+p,360-round(50*abs(a[p])/sampel));
      SetLineStyle(SolidLn,0,NormWidth);
    end;
  end;
end;
if p >= 579 then
begin
  aktif_IRQ7;      {enable IRQ7}
  repeat
  rumus_gain_high;
  rumus_gain_low;
  gain_dB;
  setcolor(10);
  tulis_gain;
until sampling = 1;
data_max[max] := 0;
data_min[min] := 0;
for p := 0 to sampel do {mengubah nilai bit ADC ke bilangan
                        kompleks FFT}
begin
  a[p].re := ((10*data3[p])/1688)*(1-cos(2*pi*(p)/sampel));
  a[p].im := 0;
end;
bit_reverse;
fft;

```

```

clearviewport;
area;
p := 0;
end;
if tombol = #13 then
begin
aktif_IRQ7; {enable IRQ7}
tombol := #0;
repeat
delay(1);
rumus_gain_high;
rumus_gain_low;
gain_dB;
setcolor(10);
tuliskan_gain;
repeat
inc(p);
if sampling = 1 then {menampilkan data ke layar monitor PC}
data_max[max] := 0;
data_min[min] := 0;
begin
putpixel(50+p,196-round(10*gain[p]),10);
setcolor(10);
moveto(50+p,30+(data3[p-1]) div 80);
lineto(50+p,30+(data3[p]) div 80);
if p < 579 shl 1 then {menampilkan data yang telah
diubah ke bilangan kompleks FFT}
begin
setcolor(14);
moveto(50+p,360-round(50*abs(a[p-1])/sampel));
lineto(50+p,360-round(50*abs(a[p])/sampel));
end;
end;
until p = 579;
tombol := readkey;
until tombol = #13;
tombol := #0;
matikan_IRQ7; {disable IRQ7}
end;
if keypressed then tombol := readkey;
until tombol = #27;
closegraph;
int_kembali;
end.

```

EVALUATION KIT AVAILABLE

MAXIM

500kps, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

General Description

The MAX120 and MAX122 complete, BiCMOS, sampling 12-bit analog-to-digital converters (ADCs) combine an on-chip track/hold (T/H) and a low-drift voltage reference with fast conversion speeds and low power consumption. The T/H's 350ns acquisition time combined with the MAX120's 1.6µs conversion time results in throughput rates as high as 500k samples per second (kpsps). Throughput rates of 333kpsps are possible with the 2.6µs conversion time of the MAX122.

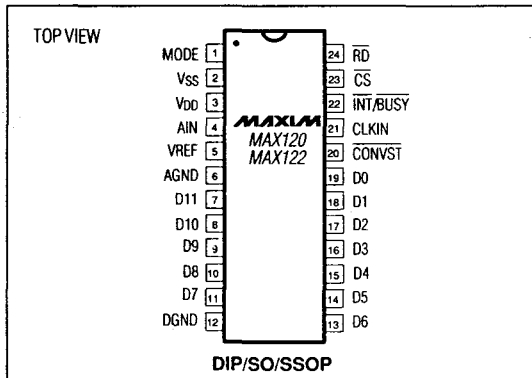
The MAX120/MAX122 accept analog input voltages from -5V to +5V. The only external components needed are decoupling capacitors for the power-supply and reference voltages. The MAX120 operates with clocks in the 0.1MHz to 8MHz frequency range. The MAX122 accepts 0.1MHz to 5MHz clock frequencies.

The MAX120/MAX122 employ a standard microprocessor (µP) interface. Three-state data outputs are configured to operate with 12-bit data buses. Data-access and bus-release timing specifications are compatible with most popular µPs without resorting to wait states. In addition, the MAX120/MAX122 can interface directly to a first in, first out (FIFO) buffer, virtually eliminating µP interrupt overhead. All logic inputs and outputs are TTL/CMOS compatible. For applications requiring a serial interface, refer to the new MAX121.

Applications

- Digital-Signal Processing
- Audio and Telecom Processing
- Speech Recognition and Synthesis
- High-Speed Data Acquisition
- Spectrum Analysis
- Data Logging Systems

Pin Configuration



Features

- ◆ 12-Bit Resolution
- ◆ No Missing Codes Over Temperature
- ◆ 20ppm/°C -5V Internal Reference
- ◆ 1.6µs Conversion Time/500kpsps Throughput (MAX120)
- ◆ 2.6µs Conversion Time/333kpsps Throughput (MAX122)
- ◆ Low Noise and Distortion:
70 dB Min SINAD;
-77 dB Max THD (MAX122)
- ◆ Low Power Dissipation: 210mW
- ◆ Separate Track/Hold Control Input
- ◆ Continuous-Conversion Mode Available
- ◆ ±5V Input Range, Overvoltage Tolerant to ±15V
- ◆ 24-Pin Narrow DIP, Wide SO and SSOP Packages

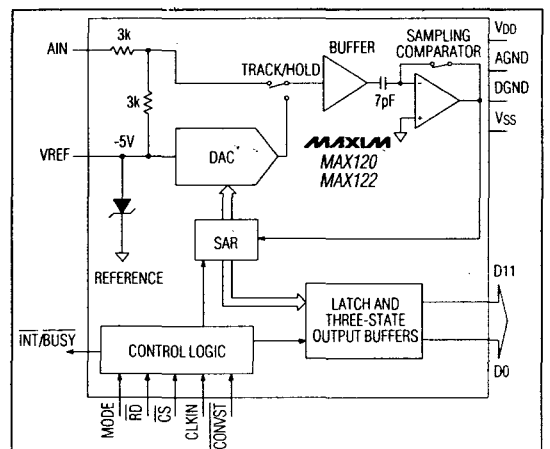
Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE	INL (LSBs)
MAX120CNG	0°C to +70°C	24 Narrow Plastic DIP	±1
MAX120CWG	0°C to +70°C	24 Wide SO	±1
MAX120CAG	0°C to +70°C	24 SSOP	±1
MAX120C/D	0°C to +70°C	Dice*	±1
MAX120ENG	-40°C to +85°C	24 Narrow Plastic DIP	±1
MAX120EWG	-40°C to +85°C	24 Wide SO	±1

Ordering Information continued on last page.

*Contact factory for dice specifications.

Functional Diagram



MAXIM

Maxim Integrated Products 1

Call toll free 1-800-998-8800 for free samples or literature.

500ksp/s, 12-Bit ADCs with Track/Hold and Reference

ABSOLUTE MAXIMUM RATINGS

V _{DD} to DGND	-0.3V to +6V
V _{SS} to DGND	+0.3V to -17V
A _{IN} to AGND	±15V
AGND to DGND	±0.3V
Digital Inputs/Outputs to DGND	-0.3V to (V _{DD} + 0.3V)
Continuous Power Dissipation (T _A = +70°C)	
Narrow Plastic DIP (derate 13.33mW/°C above +70°C)	1067mW
SO (derate 11.76mW/°C above +70°C)	941mW
SSOP (derate 8.00mW/°C above +70°C)	640mW
Narrow CERDIP (derate 12.50W/°C above +70°C)	1000mW

Operating Temperature Ranges:

MAX12_C__	0°C to +70°C
MAX12_F__	-40°C to +85°C
MAX12_MRG	-55°C to +125°C
Storage Temperature Range	-65°C to +160°C
Lead Temperature (soldering, 10 sec)	+300°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(V_{DD} = +4.75V to +5.25V, V_{SS} = -10.8V to -15.75V, I_{CLK} = 8MHz for MAX120 and 5MHz for MAX122, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
ACCURACY						
Resolution	RES		12			Bits
Differential Nonlinearity (Note 1)	DNL	12-bit no missing codes over temp. range	MAX122AC/AE		±3/4	LSB
			MAX120C/E, MAX122BC/BE/BM		±1	
		11-bit no missing codes over temp. range	MAX120M		±2	
Integral Nonlinearity (Note 1)	INL		MAX122AC/AE		±3/4	LSB
			MAX120C/E, MAX122BC/BE/BM		±1	
			MAX120M		±2	
Bipolar Zero Error (Note 1)		Code 00.00 to 00.01 transition, near A _{IN} = 0V			±3	LSB
		Temperature drift			±0.005	LSB/°C
Full-Scale Error (Notes 1, 2)		Including reference; adjusted for bipolar zero error; T _A = +25°C			±8	LSB
Full-Scale Temperature Drift		Excluding reference			±1	ppm/°C
Power-Supply Rejection Ratio (Change in FS, Note 3)	PSRR	V _{DD} only, 5V ±5%			±1/4	LSB
		V _{SS} only, -12V ±10%			±1/4	
		V _{SS} only, -15V ±5%			±1/4	
ANALOG INPUT						
Input Range			-5	5		V
Input Current		A _{IN} = +5V (approximately 6kΩ to REF)			2.5	mA
Input Capacitance (Note 4)					10	pF
Full-Power Input Bandwidth					1.5	MHz
REFERENCE						
Output Voltage		No external load, A _{IN} = 5V, T _A = +25°C	-5.02	-4.98		V
External Load Regulation		0mA < I _{SINK} < 5mA, A _{IN} = 0V			5	mV
Temperature Drift (Note 5)		MAX12_C/E			±25	ppm/°C
		MAX12_M			±30	

500kps, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

ELECTRICAL CHARACTERISTICS (continued)

(V_{DD} = +4.75V to +5.25V, V_{SS} = -10.8V to -15.75V, f_{CLK} = 8MHz for MAX120 and 5MHz for MAX122, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
DYNAMIC PERFORMANCE (MAX120: f_S = 500kHz, A_{IN} = ±5Vp-p, 100kHz; MAX122: f_S = 333kHz, A_{IN} = ±5Vp-p, 50kHz)							
Signal-to-Noise Plus Distortion	SINAD	T _A = +25°C	MAX120, MAX122	70	72		dB
			MAX122AC/AE	70			
			MAX122BC/BE/BM	69			
Total Harmonic Distortion (First Five Harmonics)	THD	T _A = +25°C	MAX120		-82	-77	dB
			MAX122		-85	-78	
			MAX122AC/AE			-77	
			MAX122BC/BE/BM			-75	
Spurious-Free Dynamic Range	SFDR	T _A = +25°C	MAX120	77	82		dB
			MAX122	78	85		
			MAX122AC/AE	77			
			MAX122BC/BE/BM	75			
CONVERSION TIME							
Synchronous	t _{CONV}	13t _{CLK}	MAX120			1.63	μs
			MAX122			2.60	
Clock Frequency	f _{CLK}		MAX120	0.1		8	MHz
			MAX122	0.1		5	
DIGITAL INPUTS (CLKIN, CONVST, RD, CS)							
Input High Voltage	V _{IH}			2.4			V
Input Low Voltage	V _{IL}					0.8	V
Input Capacitance (Note 4)						10	pF
Input Current		V _{IN} = 0V or V _{DD}				±5	μA
DIGITAL OUTPUTS (INT/BUSY, D11-D0)							
Output Low Voltage	V _{OL}	I _{SINK} = 1.6mA				0.4	V
Output High Voltage	V _{OH}	I _{SOURCE} = 1mA		V _{DD} - 0.5			V
Leakage Current	I _{LKG}	V _{IN} = 0V or V _{DD} , D11-D0				±5	μA
Output Capacitance (Note 4)						10	pF
POWER REQUIREMENTS							
Positive Supply Voltage	V _{DD}	Guaranteed by supply rejection test		4.75		5.25	V
Negative Supply Voltage	V _{SS}	Guaranteed by supply rejection test		-10.80		-15.75	V
Positive Supply Current (Note 6)	I _{DD}	V _{DD} = 5.25V, V _{SS} = -15.75V, A _{IN} = 0V			9	15	mA
Negative Supply Current (Note 6)	I _{SS}	V _{DD} = 5.25V, V _{SS} = -15.75V, A _{IN} = 0V			14	20	mA
Power Dissipation (Note 6)		V _{DD} = 5V, V _{SS} = -12V, A _{IN} = 0V			210	315	mW

Note 1: These tests are performed at V_{DD} = 5V, V_{SS} = -15V. Operation over supply is guaranteed by supply rejection tests.

Note 2: Ideal full-scale transition is at +5V - 3/2LSB = +4.9963V, adjusted for offset error.

Note 3: Supply rejection defined as change in full-scale transition voltage with the specified change in supply voltage = (FS at nominal supply) - (FS at nominal supply ± tolerance), expressed in LSBs.

Note 4: For design guidance only, not tested.

Note 5: Temperature drift is defined as the change in output voltage from +25°C to T_{MIN} or T_{MAX}. It is calculated as

$$TC = (\Delta V_{REF}/V_{REF})/(\Delta T)$$

Note 6: CS = RD = CONVST = 0V, MODE = 5V

500kps, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

TIMING CHARACTERISTICS

(VDD = +5V, VSS = -12V to -15V, 100% tested, TA = TMIN to TMAX, unless otherwise noted.) (Note 7)

PARAMETER	SYMBOL	CONDITIONS	TA = +25°C			MAX12_C/E			MAX12_M			UNITS
			MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
CS to RD Setup Time	tCS		0			0			0			ns
CS to RD Hold Time	tCH		0			0			0			ns
CONVST Pulse Width	tCW		30			30			30			ns
RD Pulse Width	tRW		tDA			tDA			tDA			ns
Data-Access Time	tDA	CL = 100pF	40	75			100			120		ns
Bus-Relinquish Time	tDH		30	50			65			80		ns
RD or CONVST to BUSY	tB0	CL = 50pF	30	75			100			120		ns
CLKIN to BUSY or INT	tB1	CL = 50pF	70	110			150			180		ns
CLKIN to BUSY Low	tB2	In mode 5	45	90			120			150		ns
RD to INT High	tIH	CL = 50pF	30	50			75			90		ns
BUSY or INT to Data Valid	tBD	CL (Data) = 100pF CL (INT, BUSY) = 50pF		20			30			35		ns
Acquisition Time (Note 8)	tAQ		350			350			400			ns
Aperture Delay (Note 8)	tAP		10									ns
Aperture Jitter (Note 8)			30									ps

Note 7: Control inputs specified with $t_r = t_f = 5\text{ns}$ (10% to 90% of +5V) and timed from a 1.6V voltage level. Output delays are measured to +0.8V if going low, or +2.4V if going high. For bus-relinquish time, a change of 0.5V is measured. See Figures 1 and 2 for load circuits.

Note 8: For design guidance only, not tested.

Pin Description

PIN	NAME	FUNCTION
1	MODE	Mode Input - hard-wire to set operational mode. VDD: Single conversion, INT Output OPEN: Single conversion, BUSY Output DGND: Continuous conversions, BUSY Output
2	VSS	Negative Power Supply, -12V or -15V
3	VDD	Positive Power Supply, +5V
4	AIN	Sampling Analog Input, ±5V bipolar input range
5	VREF	-5V Reference Output - bypass to AGND with 22μF 0.1μF

500kps, 12-Bit ADCs with Track/Hold and Reference

Pin Description (continued)

PIN	NAME	FUNCTION
6	AGND	Analog Ground
7-11, 13-19	D11-D0	Three-State Data Outputs D11 (MSB) to D0 (LSB)
12	DGND	Digital Ground
20	$\overline{\text{CONVST}}$	Convert Start Input initiates conversions on its falling edge.
21	CLKIN	Clock Input. Drive with TTL-compatible clock from 0.1MHz to 8MHz (MAX120), 0.1MHz to 5MHz (MAX122).
22	$\overline{\text{INT}}/\text{BUSY}$	$\overline{\text{Interrupt}}$ or $\overline{\text{Busy}}$ Output indicates converter status. If MODE is connected to V_{DD} , configure for an $\overline{\text{INT}}$ output. If MODE is open or connected to DGND, configure for a BUSY output. See operational diagrams.
23	$\overline{\text{CS}}$	Chip Select Input - active low. When $\overline{\text{RD}}$ is low, enables the three-state outputs. If $\overline{\text{CONVST}}$ and $\overline{\text{RD}}$ are low, a conversion is initiated on the falling edge of $\overline{\text{CS}}$.
24	$\overline{\text{RD}}$	Read Input - active low. When $\overline{\text{CS}}$ is low, $\overline{\text{RD}}$ enables the three-state outputs. If $\overline{\text{CONVST}}$ and $\overline{\text{CS}}$ are low, a conversion is initiated on the falling edge of $\overline{\text{RD}}$.

Detailed Description

ADC Operation

The MAX120/MAX122 use successive approximation and input T/H circuitry to convert an analog signal to a series of 12-bit digital-output codes. The control logic interfaces easily to most μPs , requiring only a few passive components for most applications. The T/H does not require an external capacitor. Figure 3 shows the MAX120/MAX122 in the simplest operational configuration.

Analog Input Track/Hold

Figure 4 shows the equivalent input circuit, illustrating the sampling architecture of the ADC's analog comparator. An internal buffer charges the hold capacitor to minimize the required acquisition time between conversions. The analog input appears as a $6\text{k}\Omega$ resistor in parallel with a 10pF capacitor.

Between conversions, the buffer input is connected to AIN through the input resistance. When a conversion starts, the buffer input disconnects from AIN, thus sampling the input. At the end of the conversion, the buffer input reconnects to AIN, and the hold capacitor once again charges to the input voltage.

The T/H is in tracking mode whenever a conversion is NOT in progress. Hold mode starts approximately 10ns after a conversion is initiated. Variation in this delay from one conversion to the next (aperture jitter) is typically 30ps . Figures 7 through 11 detail the T/H mode and interface timing for the various interface modes.

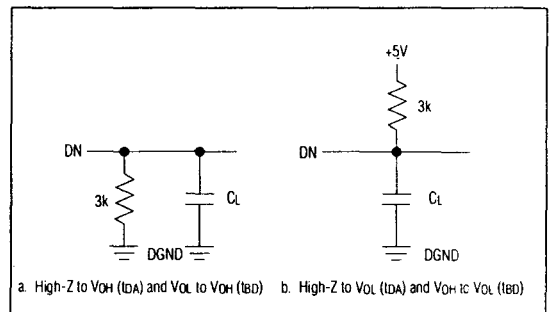


Figure 1. Load Circuits for Access Time

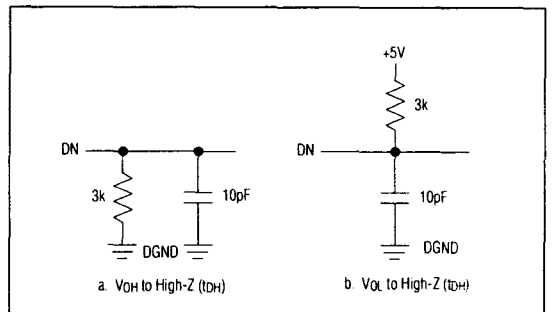


Figure 2. Load Circuits for Bus-Relinquish Time

500kps, 12-Bit ADCs with Track/Hold and Reference

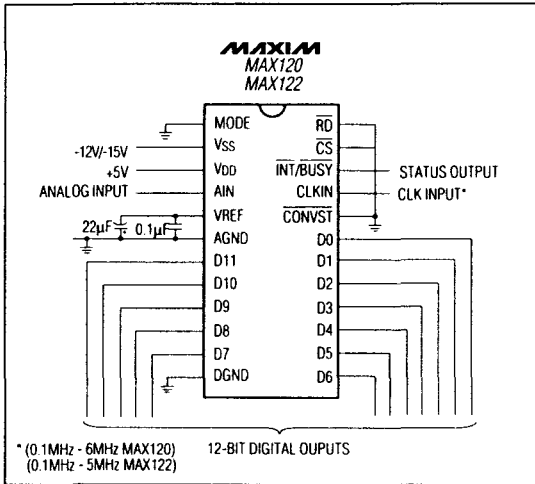


Figure 3. MAX120/MAX122 in the Simplest Operational Mode (Continuous Conversion)

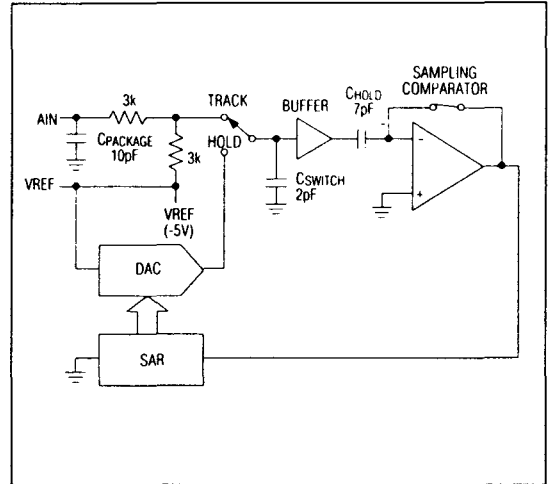


Figure 4. Equivalent Input Circuit

Internal Reference

The MAX120/MAX122 -5.00V buried-zener reference biases the internal DAC. The reference output is available at the VREF pin and must be bypassed to the AGND pin with a 0.1µF ceramic capacitor in parallel with a 22µF or greater electrolytic capacitor. The electrolytic capacitor's equivalent series resistance (ESR) must be 100mΩ or less to properly compensate the reference output buffer. Sanyo's organic semiconductor works well.

Sanyo Video Components (USA)
Phone: (619) 661-6835 FAX: (619) 661-1055

Sanyo Electric Company, LTD. (Japan)
Phone: 0720-70-1005 FAX: 0720-70-1174

Sanyo Fisher Vertriebs GmbH (Germany)
Phone: 06102-27041, ext. 44 FAX: 06102-27045

Proper bypassing minimizes reference noise and maintains a low impedance at high frequencies. The internal reference output buffer can sink up to a 5mA external load.

An external reference voltage can be used to overdrive the MAX120/MAX122's internal reference if it ranges from -5.05V to -5.10V and is capable of sinking a minimum of 5mA. The external VREF bypass capacitors are still required.

Digital Interface

External Clock

The MAX120/MAX122 require a TTL-compatible clock for proper operation. The MAX120 accepts clocks in the 0.1MHz to 8MHz frequency range when operating in modes 1-4 (see *Operating Modes* section). The maximum clock frequency is limited to 6MHz when operating in mode 5. The MAX122 requires a 0.1MHz to 5MHz clock for operation in all five modes. The minimum clock frequency for both the MAX120 and MAX122 is limited to 0.1MHz, due to the T/H's droop rate.

Clock and Control Synchronization

If the clock and convert start inputs (CONVST or \overline{RD} and \overline{CS} - see *Operating Modes* section) are not synchronized, the conversion time can vary from 13 to 14 clock cycles. The successive approximation register (SAR) always changes state on the CLKIN input's rising edge. To ensure a fixed conversion time, refer to Figure 5 and the following guidelines.

For a conversion time of 13 clock cycles, the convert start input(s) should go low at least 50ns before CLKIN's next rising edge. For a conversion time of 14 clock cycles, the convert start input(s) should go low within 10ns of CLKIN's next rising edge. If the convert start input(s) go low from 10ns to 50ns before CLKIN's next rising edge, the number of clock cycles required is undefined and can be either 13 or 14. For best analog performance, synchronize the convert start inputs with the clock input.

500kps, 12-Bit ADCs with Track/Hold and Reference

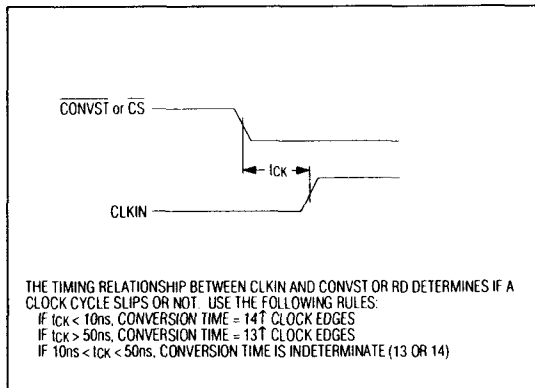


Figure 5. Clock and Control Synchronization

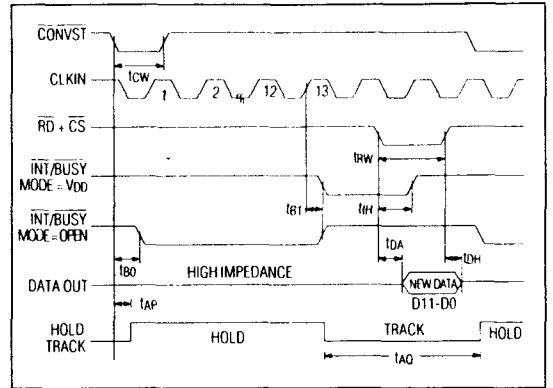


Figure 7. Full-Control Mode (Mode 1)

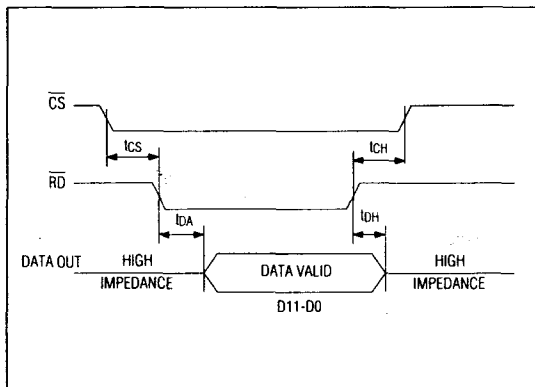


Figure 6. Data-Access and Bus-Relinquish Timing

Output Data Format

The conversion result is output on a 12-bit data bus with a 75ns data-access time. The output data format is two's-complement. The three input control signals ($\overline{\text{CS}}$, $\overline{\text{RD}}$, and $\overline{\text{CONVST}}$), the $\overline{\text{INT/BUSY}}$ converter status output, and the 12 bits of output data can interface directly to a 16-bit data bus. See Figure 6 for data-access timing.

Timing and Control

The MAX120/MAX122 have five operational modes as outlined in Figures 7-11 and discussed in the *Operating Modes* section.

Full-control mode (mode 1) provides maximum control to the user for convert start and data-read operations.

Full-control mode is for μPs with or without wait-state capability. Stand-alone mode (mode 2) and continuous-conversion mode (mode 5) are for systems without μPs , or for μP -based systems where the ADC and the μP are linked through first in, first out (FIFO) buffers or direct memory access (DMA) ports. Slow-memory mode (mode 3) is intended for μPs that can be forced into a wait state during the ADC's conversion time. ROM mode (mode 4) is for μPs that cannot be forced into a wait state.

In all five operating modes, the start of a conversion is controlled by one of three digital inputs: $\overline{\text{CONVST}}$, $\overline{\text{RD}}$, or $\overline{\text{CS}}$. Figure 12 shows the logic equivalent for the conversion circuitry. In any operating mode, $\overline{\text{CONVST}}$ must be low for a conversion to occur. Once the conversion is in progress, it cannot be restarted.

Read operations are controlled by $\overline{\text{RD}}$ and $\overline{\text{CS}}$. Both of these digital inputs must be low to read output data. The $\overline{\text{INT/BUSY}}$ output indicates the converter's status and determines when the data from the most recent conversion is available. The MODE input configures the $\overline{\text{INT/BUSY}}$ output as follows:

If $\text{MODE} = \text{VDD}$, $\overline{\text{INT/BUSY}}$ functions as an INTERRUPT output. In this configuration, $\overline{\text{INT/BUSY}}$ goes low when the conversion is complete and returns high after the conversion data has been read.

If MODE is left open or tied to DGND , $\overline{\text{INT/BUSY}}$ functions as a BUSY output. In this case, $\overline{\text{INT/BUSY}}$ goes low at the start of a conversion and remains low until the conversion is complete and the data is available at D0-D11.

500kps, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

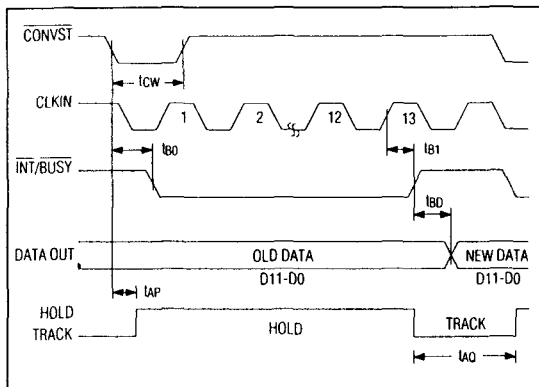


Figure 8. Stand-Alone Mode (Mode 2)

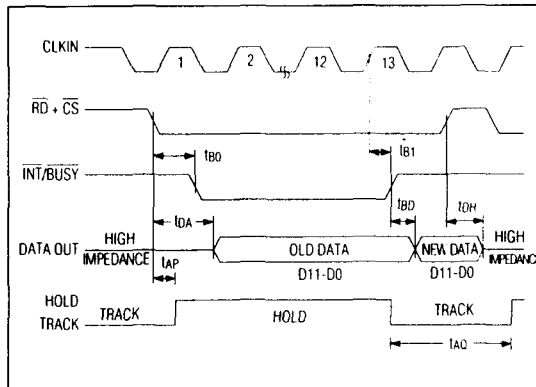


Figure 9. Slow-Memory Mode (Mode 3)

Initialization After Power-Up

On power-up, the first MAX120/MAX122 conversion is valid if the following conditions are met:

- 1) Allow 14 clock cycles for the internal T/H to enter the track mode, plus a minimum of 350ns in the track mode for the data-acquisition time.
- 2) Make sure the reference voltage has settled. Allow 0.5ms for each 1 μ F of reference bypass capacitance (11ms for a 22 μ F capacitor).

Operating Modes

Mode 1: (Full-Control Mode)

Figure 7 shows the timing diagram for full-control mode (mode 1). In this mode, the μ P controls the conversion-start and data-read operations independently.

A falling edge on $\overline{\text{CONVST}}$ places the T/H into hold mode and starts a conversion in the SAR. The conversion is complete in 13 or 14 clock cycles as discussed in the *Clock and Control Synchronization* section. A change in the $\overline{\text{INT/BUSY}}$ output state signals the end of a conversion as follows:

If $\text{MODE} = \text{VDD}$, the end of conversion is signaled by the $\overline{\text{INT/BUSY}}$ output falling edge.

If $\text{MODE} = \text{OPEN}$ or DGND , the $\overline{\text{INT/BUSY}}$ output goes low while the conversion is in progress and returns high when the conversion is complete.

When the conversion is complete, the data can be read without initiating a new conversion by pulling $\overline{\text{RD}}$ and $\overline{\text{CS}}$ low and leaving $\overline{\text{CONVST}}$ high. To start a new conversion without reading data, $\overline{\text{RD}}$ and $\overline{\text{CS}}$ should remain high while $\overline{\text{CONVST}}$ is driven low. To simultaneously read data and initiate a new conversion, $\overline{\text{CONVST}}$, $\overline{\text{RD}}$, and $\overline{\text{CS}}$ should all be pulled low. Note: Allow at least 350ns for T/H acquisition time between the end of one conversion and the beginning of the next.

Mode 2: Stand-Alone Operation ($\text{MODE} = \text{OPEN}$, $\overline{\text{RD}} = \overline{\text{CS}} = \text{DGND}$)

For systems that do not use or require full-bus interfacing, the MAX120/MAX122 can be operated in stand-alone mode directly linked to memory through DMA ports or a FIFO buffer. In stand-alone mode, a conversion is initiated by a falling edge on $\overline{\text{CONVST}}$. The data outputs are always enabled; data changes at the end of a conversion as indicated by a rising edge on $\overline{\text{INT/BUSY}}$. See Figure 8 for stand-alone mode timing.

Mode 3: Slow-Memory Mode ($\overline{\text{CONVST}} = \text{GND}$, $\text{MODE} = \text{OPEN}$)

Taking $\overline{\text{RD}}$ and $\overline{\text{CS}}$ low places the T/H into hold mode and starts a conversion. $\overline{\text{INT/BUSY}}$ remains low while the conversion is in progress and can be used as a wait input to the μ P. Data from the previous conversion appears on the data bus until the conversion end is indicated by $\overline{\text{INT/BUSY}}$. See Figure 9 for slow-memory mode timing.

500kps, 12-Bit ADCs with Track/Hold and Reference

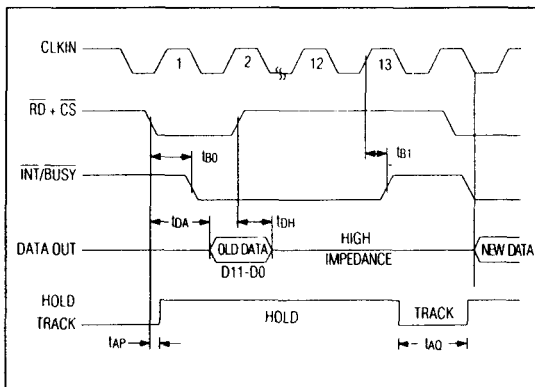


Figure 10. ROM Mode (Mode 4)

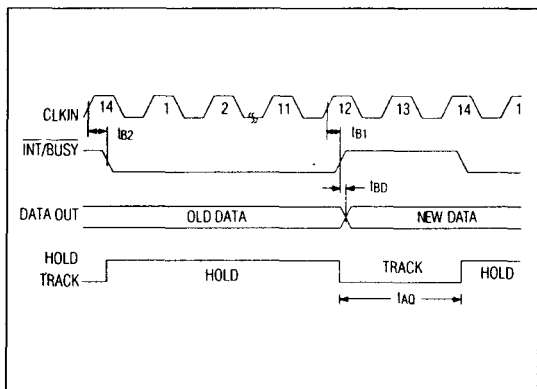


Figure 11. Continuous-Conversion Mode (Mode 5)

Mode 4: ROM Mode (MODE = OPEN, CONVST = GND)

In ROM mode, the MAX120/MAX122 behave like a fast-access memory location and avoid placing the μ P into a wait state. Pulling \overline{RD} and \overline{CS} low places the T/H in hold mode, starts a conversion, and reads data from the previous conversion. Data from the first read in a sequence is often disregarded when this interface mode is used. A second read operation accesses the first conversion's result and also starts a new conversion. The time between successive read operations must be longer than the sum of the T/H acquisition time and the MAX120/MAX122 conversion time. See Figure 10 for ROM-mode timing.

Mode 5: Continuous-Conversion Mode (CONVST = \overline{RD} = \overline{CS} = MODE = GND)

For systems that do not use or require full-bus interfacing, the MAX120/MAX122 can operate in continuous-conver-

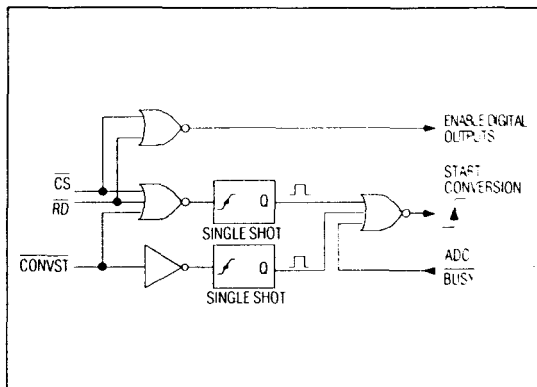


Figure 12. Conversion-Control Logic

sion mode, directly linked to memory through DMA ports or a FIFO buffer. In this mode, conversions are performed continuously at the rate of one conversion for every 14 clock cycles, which includes 2 clock cycles for the T/H acquisition time. To satisfy the 350ns minimum acquisition time requirement within 2 clock cycles, the MAX120's maximum clock frequency is 6MHz when operating in mode 5.

The data outputs are always enabled and "new" data appears on the output bus at the end of a conversion as indicated by the INT/BUSY output rising edge. The MODE input should be hard-wired to GND. Pulling \overline{CS} , \overline{RD} , or CONVST high stops conversions. See Figure 11 for continuous-conversion mode timing.

Applications Information

Using FIFO Buffers

Using FIFO memory to buffer blocks of data from the MAX120 reduces μ P interrupt overhead time by enabling the μ P to process data while the MAX120, unassisted, writes conversion results to the FIFO. To retrieve a block of data, the μ P reads from the FIFO via a read-interrupt cycle. Read and write operations for the FIFO are completely asynchronous. Figure 13 shows the MAX120 operating in continuous-conversion mode (mode 5), writing data directly into the two IDT7200 256 x 9 FIFO buffers at the rate of 428kps. The μ P is interrupted to read the accumulated data by the FIFO's half-full (HF) flag approximately three times per millisecond. For operation at 500kps, use an 8MHz clock, and pulse CONVST at 500kHz. The full flag (FF) indicates that the FIFO is full. If this flag is ignored, data may be lost. If necessary, conversions can be inhibited by pulling \overline{CS} , \overline{RD} , or CONVST high. The FIFO's read cycle times are as fast as 15ns, satisfying most system speed requirements. The RESET input resets all data in the FIFO to zero.

500kps, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

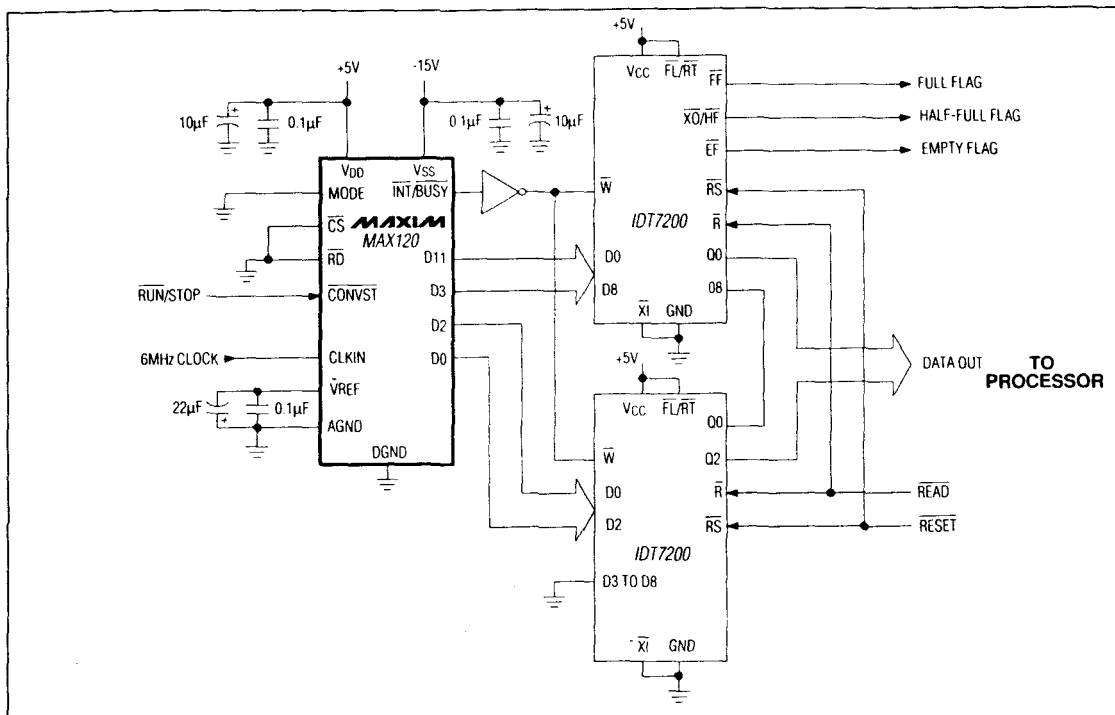


Figure 13. Using MAX120 with FIFO Memory

For synchronous operation, the $\overline{\text{CONVST}}$ pin may be used to initiate conversions, as described in the *Operating Modes* section (Mode 2: Stand-Alone Operation).

Digital-Bus Noise

If the ADC's data bus is active during a conversion, coupling from the data pins to the ADC comparator can cause errors. Using slow-memory mode (mode 3) avoids this problem by placing the μP in a wait state during the conversion. If the data bus is active during the conversion in either mode 1 or 4, use three-state drivers to isolate the bus from the ADC.

In ROM mode (mode 4), considerable digital noise is generated in the ADC when $\overline{\text{RD}}$ or $\overline{\text{CS}}$ go high, disabling the output buffers after a conversion is started. This noise can cause errors if it occurs at the same instant the SAR latches a comparator decision. To avoid this problem, $\overline{\text{RD}}$ and $\overline{\text{CS}}$ should be active for less than 1 clock cycle. If this is not possible, $\overline{\text{RD}}$ or $\overline{\text{CS}}$ should go high coinciding with CLKIN 's falling edge, since the comparator output is always latched at CLKIN 's rising edge.

Layout, Grounding, and Bypassing

For best system performance, use printed circuit boards with separate analog and digital ground planes. Wire-wrap boards are not recommended. The two ground planes should be tied together at the low-impedance power-supply source, as shown in Figure 14.

The board layout should ensure that digital and analog signal lines are kept separate from each other as much as possible. Do not run analog and digital (especially clock) lines parallel to one another.

The ADC's high-speed comparator is sensitive to high-frequency noise in the V_{DD} and V_{SS} power supplies. Bypass these supplies to the analog ground plane with 0.1 μF and 10 μF bypass capacitors. Minimize capacitor lead lengths for best noise rejection. If the +5V power supply is very noisy, connect a 5 Ω resistor, as shown in Figure 14. Figure 15 shows the negative power-supply (V_{SS}) rejection vs. frequency. Figure 16 shows the positive power-supply (V_{DD}) rejection vs. frequency, with and without the optional 5 Ω resistor.

500kps, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

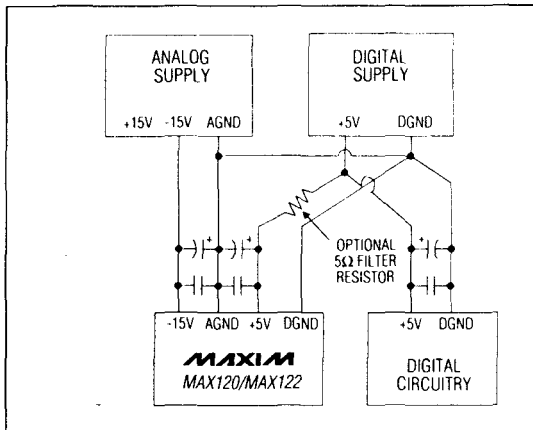


Figure 14. Power-Supply Grounding

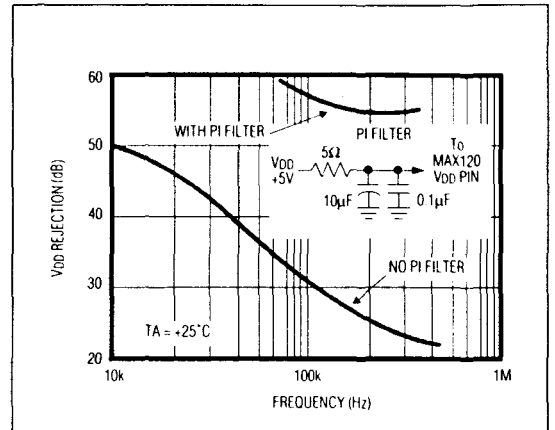


Figure 16. V_{DD} Power-Supply Rejection vs. Frequency

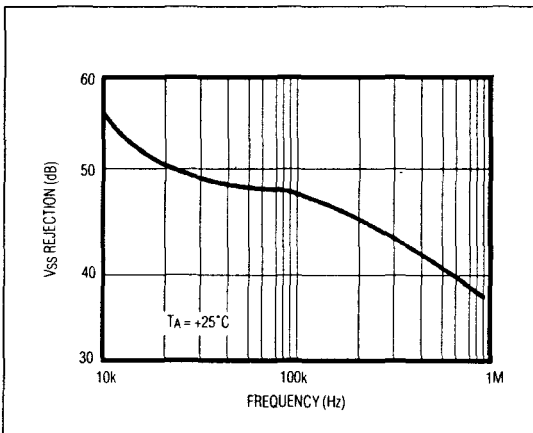


Figure 15. V_{SS} Power-Supply Rejection vs. Frequency

Gain and Offset Adjustment

Figure 17 plots the bipolar input/output transfer function for the MAX120/MAX122. Code transitions occur halfway between successive integer LSB values. Output coding is two's-complement binary with $1\text{LSB} = 2.44\text{mV}$ ($10\text{V}/4096$).

In applications where gain (full-scale range) adjustment is required, Figure 18's circuit can be used. If both offset and gain (full-scale range) need adjustment, either of the circuits in Figures 19 and 20 can be used. Offset should be adjusted before gain for either of these circuits.

To adjust bipolar offset with Figure 19's circuit, apply $+1/2\text{LSB}$ (0.61mV) to the noninverting amplifier input and adjust $R4$ for output-code flicker between 0000 0000 0000 and 0000 0000 0001. For full scale, apply $\text{FS} - 1/2\text{LSB}$ (2.4988V) to the amplifier input and adjust $R2$ so the output code flickers between 0111 1111 1110 and 0111 1111 1111. There may be some interaction between these adjustments. The MAX120/MAX122 transfer function used in conjunction with Figure 19's circuit is the same as Figure 17, except the full-scale range is reduced to 2.5V .

To adjust bipolar offset with Figure 20's circuit, apply $-1/2\text{LSB}$ (-1.22mV) at V_{IN} and adjust $R5$ for output-code flicker between 0000 0000 0000 and 0000 0000 0001. For gain adjustment, apply $-\text{FS} + 1/2\text{LSB}$ (-4.9951V) at V_{IN} and adjust $R1$ so the output code flickers between 0111 1111 1110 and 0111 1111 1111. As with Figure 20's circuit, the offset and gain adjustments may interact. Figure 21 plots the transfer function for Figure 20's circuit.

Dynamic Performance

High-speed sampling capability and 500kps throughput (333kps for the MAX122) make the MAX120/MAX122 ideal for wideband-signal processing. To support these and other related applications, fast fourier transform (FFT) test techniques are used to guarantee the ADC's dynamic frequency response, distortion, and noise at the rated throughput. Specifically, this involves applying a low-distortion sine wave to the ADC input and recording the digital conversion results for a specified time. The data is then analyzed using an FFT algorithm, which determines its spectral content.

500kps, 12-Bit ADCs with Track/Hold and Reference

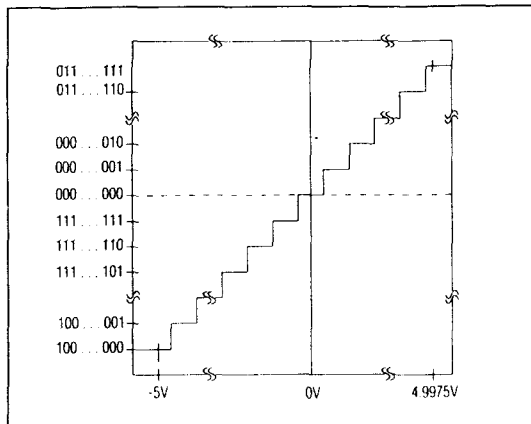


Figure 17. Bipolar Transfer Function

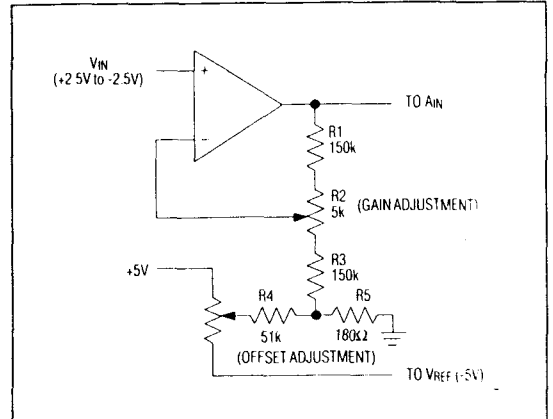


Figure 19. Offset and Gain Adjustment (Noninverting)

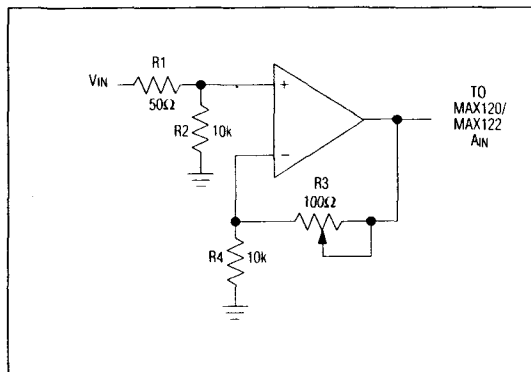


Figure 18. Trim Circuit for Gain Only

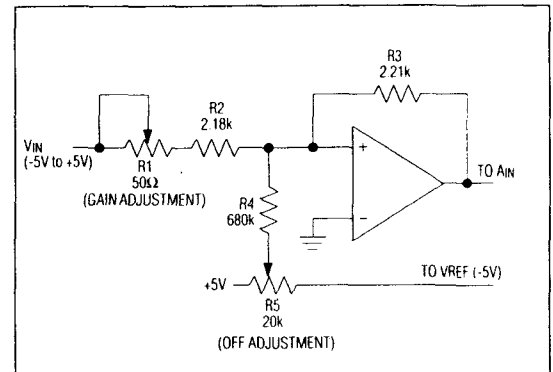


Figure 20. Offset and Gain Adjustment (Inverting)

ADCs have traditionally been evaluated by specifications such as zero and full-scale error, integral nonlinearity (INL), and differential nonlinearity (DNL). Such parameters are widely accepted for specifying performance with DC and slowly varying signals, but are less useful in signal processing applications where the ADC's impact on the system transfer function is the main concern. The significance of various DC errors does not translate well to the dynamic case, so different tests are required.

Signal-to-Noise Ratio and Effective Number of Bits

The signal-to-noise plus distortion ratio (SINAD) is the ratio of the fundamental input frequency's RMS amplitude to the RMS amplitude of all other ADC output signals. The output band is limited to frequencies above DC and below one-half the ADC sample rate.

The theoretical minimum ADC noise is caused by quantization error and is a direct result of the ADC's resolution: $SNR = (6.02N + 1.76)dB$, where N is the number of bits of resolution. A perfect 12-bit ADC can, therefore, do no better than 74dB. An FFT plot shows the output level in various spectral bands. Figure 22 shows the result of sampling a pure 100kHz sinusoid at a 500kps rate with the MAX120.

By transposing the equation that converts resolution to SNR, we can, from the measured SINAD, determine the effective resolution (or effective number of bits) the ADC provides: $N = (SINAD - 1.76)/6.02$. Figure 22 shows the effective number of bits as a function of the input frequency for the MAX120. The MAX122 performs similarly.

500kps, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

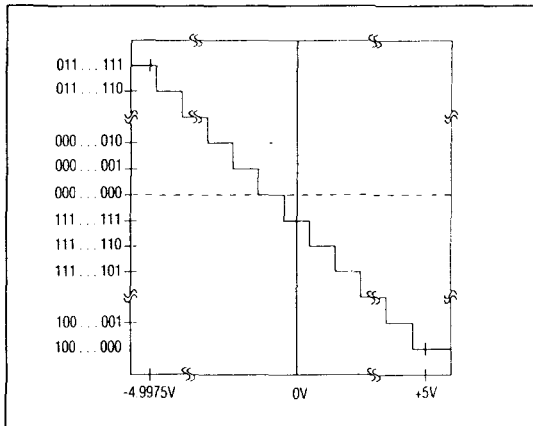


Figure 21. Inverting Bipolar Transfer Function

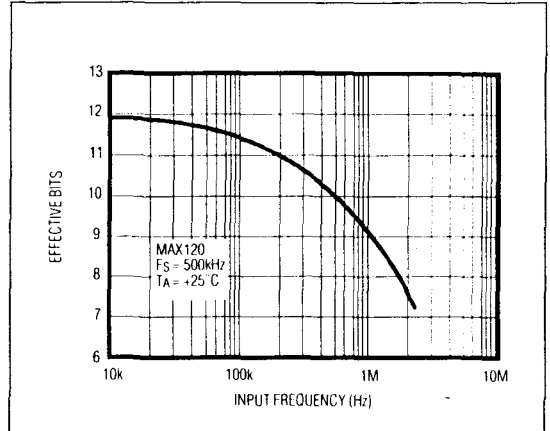


Figure 23. Effective Bits vs. Input Frequency

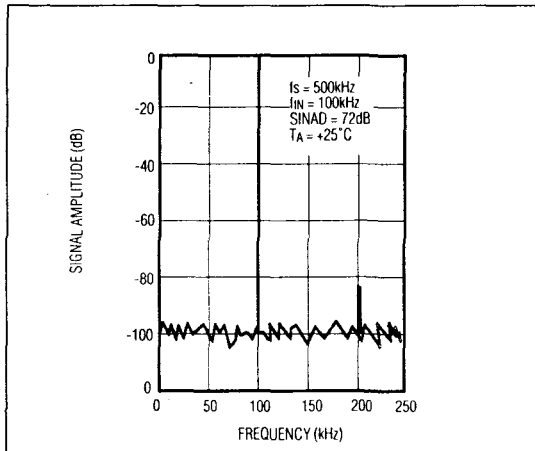


Figure 22. MAX120 FFT Plot

Total Harmonic Distortion

If a pure sine wave is sampled by an ADC at greater than the Nyquist frequency, the nonlinearities in the ADC's transfer function create harmonics of the input frequency in the sampled output data.

Total harmonic distortion (THD) is the ratio of the RMS sum of all harmonics (in the frequency band above DC and below one-half the sample rate, but not including the DC component) to the RMS amplitude of the fundamental frequency. This is expressed as follows:

$$\text{THD} = 20 \log \frac{\sqrt{V_2^2 + V_3^2 + V_4^2 + \dots + V_N^2}}{V_1}$$

where V_1 is the fundamental RMS amplitude, and V_2 to V_N are the amplitudes of the 2nd through Nth harmonics. The THD specification in the *Electrical Characteristics* table includes the 2nd through 5th harmonics.

Intermodulation Distortion

If the ADC input signal consists of more than one spectral component, the ADC transfer function nonlinearities produce intermodulation distortion (IMD) in addition to THD. IMD is the change in one sinusoidal input caused by the presence of another sinusoidal input at a different frequency.

If two pure sine waves of frequency f_a and f_b are applied to the ADC input, nonlinearities in the ADC transfer function create distortion products at sum and difference frequencies of $m f_a \pm n f_b$, where m and $n = 0, 1, 2, 3, \dots$. THD includes those distortion products with m or n equal to zero. Intermodulation distortion consists of all distortion products for which neither m nor n equal zero. For example, the 2nd-order IMD terms include $(f_a + f_b)$ and $(f_a - f_b)$ while the 3rd-order IMD terms include $(2f_a + f_b)$, $(2f_a - f_b)$, $(f_a + 2f_b)$, and $(f_a - 2f_b)$.

If the two input sine waves are equal in magnitude, the value (in decibels) of the 2nd-order IMD products can be expressed by the following formula:

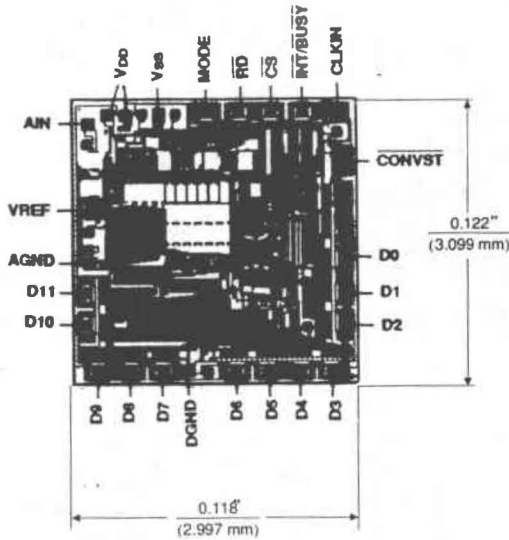
$$\text{IMD} (f_a \pm f_b) = 20 \log \left[\frac{\text{amplitude at } (f_a \pm f_b)}{\text{amplitude at } f_a} \right]$$

500ksp/s, 12-Bit ADCs with Track/Hold and Reference

Spurious-Free Dynamic Range

Spurious-free dynamic range is the ratio of the fundamental RMS amplitude to the amplitude of the next largest spectral component (in the frequency band above DC and below one-half the sample rate). Usually the next largest spectral component occurs at some harmonic of the input frequency. However, if the ADC is exceptionally linear, it may occur only at a random peak in the ADC's noise floor.

Chip Topography



MAX120/MAX122 TRANSISTOR COUNT 1920;
SUBSTRATE CONNECTED TO V_{DD}.

Ordering Information (continued)

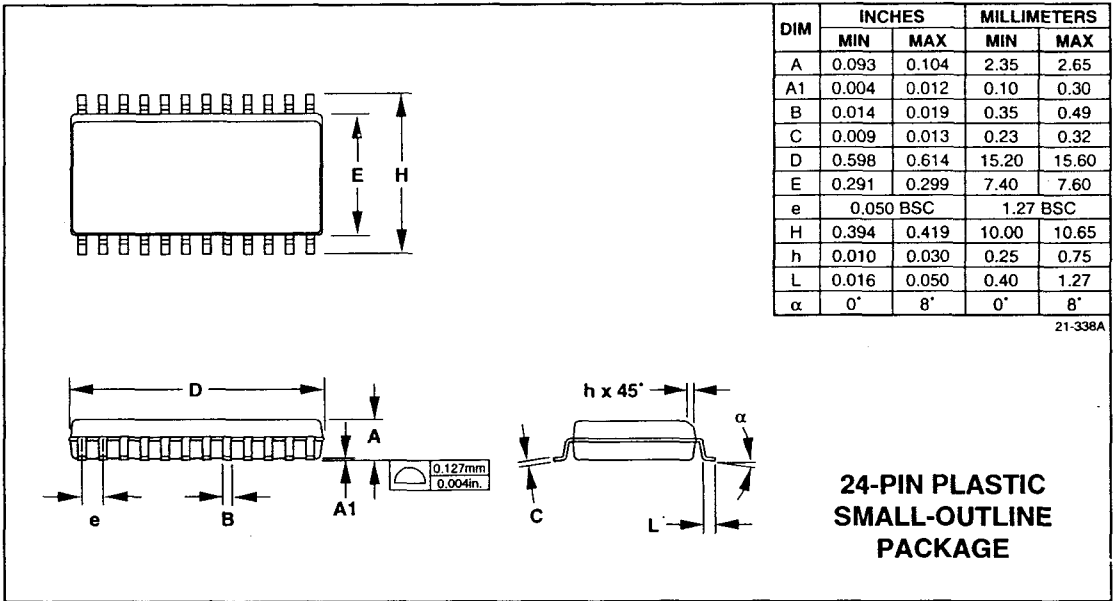
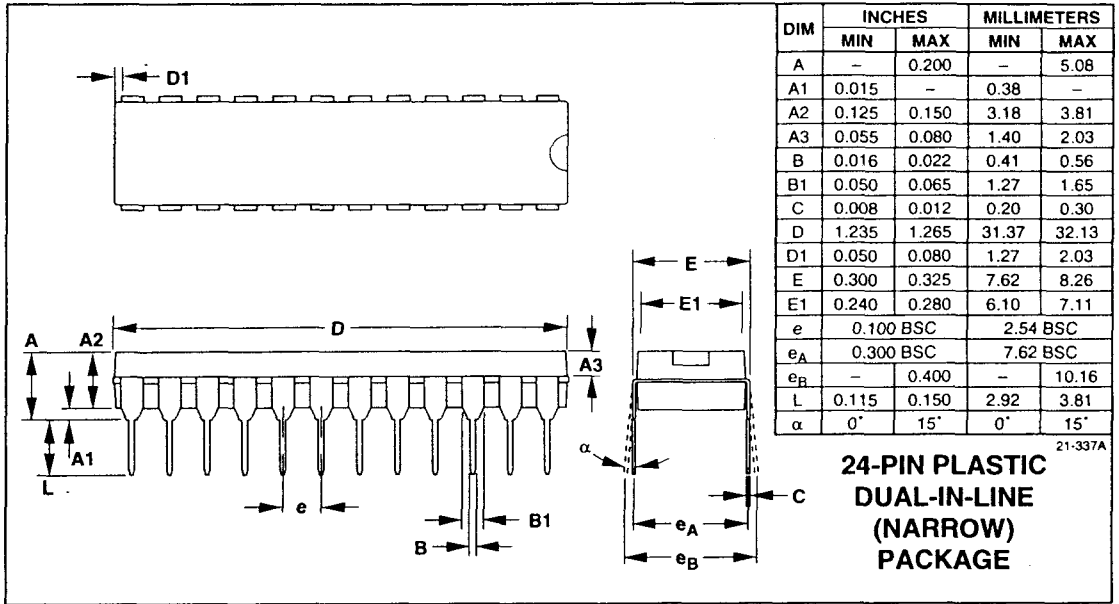
PART	TEMP. RANGE	PIN-PACKAGE	INL (LSBs)
MAX120EAG	-40°C to +85°C	24 SSOP	±1
MAX120MRG	-55°C to +125°C	24 Narrow CERDIP	±2
MAX122ACNG	0°C to +70°C	24 Narrow Plastic DIP	±3/4
MAX122BCNG	0°C to +70°C	24 Narrow Plastic DIP	±1
MAX122ACWG	0°C to +70°C	24 Wide SO	±3/4
MAX122BCWG	0°C to +70°C	24 Wide SO	±1
MAX122ACAG	0°C to +70°C	24 SSOP	±3/4
MAX122BCAG	0°C to +70°C	24 SSOP	±1
MAX122BC/D	0°C to +70°C	Dice*	±1
MAX122AENG	-40°C to +85°C	24 Narrow Plastic DIP	±3/4
MAX122BENG	-40°C to +85°C	24 Narrow Plastic DIP	±1
MAX122AEWG	-40°C to +85°C	24 Wide SO	±3/4
MAX122BEWG	-40°C to +85°C	24 Wide SO	±1
MAX122AEAG	-40°C to +85°C	24 SSOP	±3/4
MAX122BEAG	-40°C to +85°C	24 SSOP	±1
MAX122BMRG	-55°C to +125°C	24 Narrow CERDIP	±1
MAX120EVKIT-DIP†	0°C to +70°C	Plastic DIP - Through Hole	

* Contact factory for dice specifications.
† MAX120 EV kit can be used to evaluate the MAX122; when ordering the EV kit, ask for a free sample of the MAX122.

500kps, 12-Bit ADCs with Track/Hold and Reference

Package Information

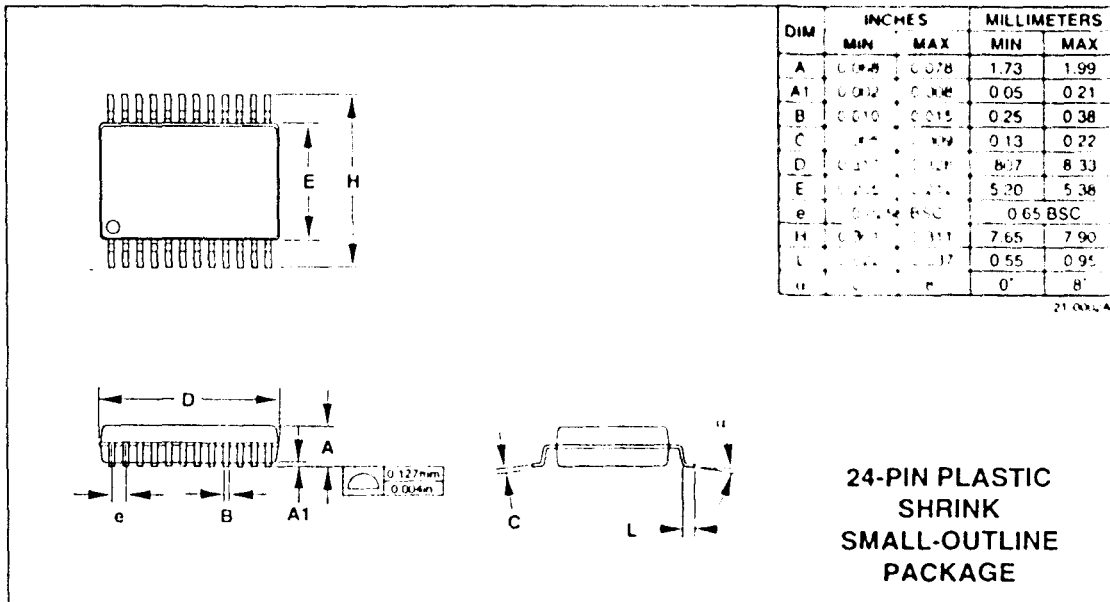
MAX120/MAX122



500ksp/s, 12-Bit ADCs with Track/Hold and Reference

MAX120/MAX122

Package Information (continued)

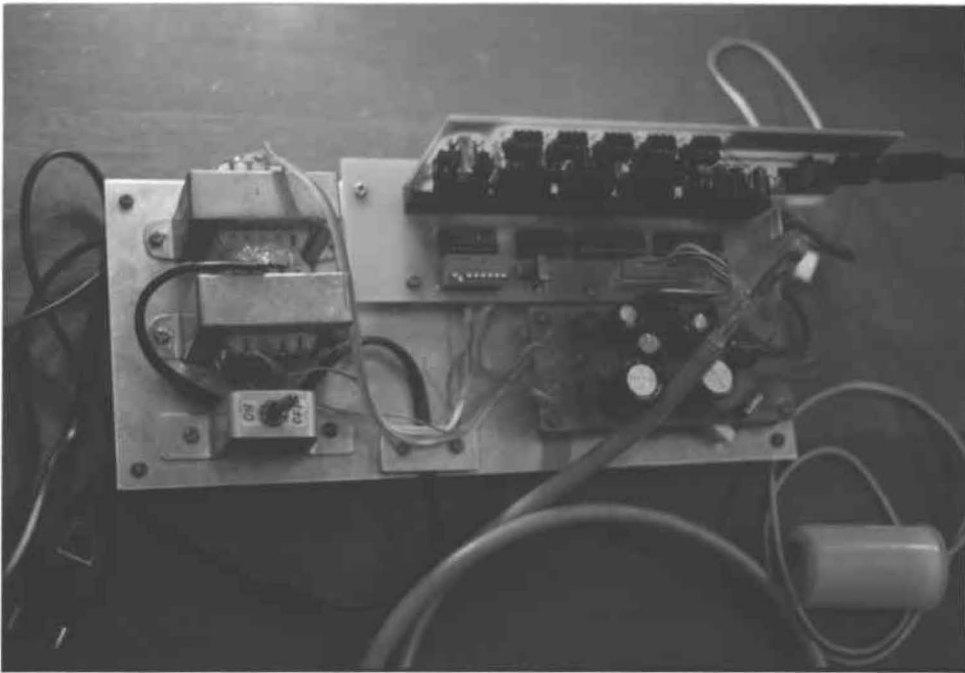


Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

16 Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 (408) 737-7600

© 1994 Maxim Integrated Products Printed USA MAXIM is a registered trademark of Maxim Integrated Products.

LAMPIRAN E



BIODATA



Nama Lengkap : **Handoko Djoko**
NRP : 5103096040
NIRM : 96.7.003.31073.44919
Tempat, Tanggal Lahir : Sampit, 14 Mei 1977
Agama : Budha
Alamat Rumah : Rangkas V/61 Sampit
Kalimantan - Tengah

Riwayat Pendidikan :

- SD Katolik Yos Soedarso Sampit, lulus tahun 1990.
- SMP Katolik Sampit, lulus tahun 1993.
- SMA Katolik Taruna Jaya Sampit, lulus tahun 1996.
- Universitas Katolik Widya Mandala Fakultas Teknik Jurusan Teknik Elektro, Surabaya, angkatan 1996.