

The PDSP16116A has been designed with an integral Block Floating Point system which can be used, in conjunction with other Mitel Semiconductor PDSP parts, to process FFTs with a combination of speed and accuracy previously unobtainable. All the functionality of this BFP system is contained within the PDSP parts, which are designed to interface easily to achieve a powerful FFT solution.

A butterfly processor based on the 20MHz PDSP16116A will allow the following FFT benchmarks:

- 1024 point complex radix-2 transform in 259us
- 512 point complex radix-2 transform in 118us
- 256 point complex radix-2 transform in 53us

This compares favourably with the current industry standard benchmark of around 2ms for a 1024 point complex FFT, but if speed is all important for a particular application, then the Mitel Semiconductor's PDSP16112/A 16x12 Complex Multiplier can double the PDSP16116/A performance with up to 70dB of dynamic range.

The FFT Algorithm

The Fast Fourier Transform is essentially a computationally efficient algorithm for extracting spectral information from signal waveforms, which may be in real time or recorded form (i.e. a transformation from the time domain to the frequency domain). It is often used to dramatic effect in a growing range of applications including radar and sonar processing, speech recognition and image processing. It is no less accurate than the related Discrete Fourier Transform (DFT), but it enjoys a vastly improved performance due to the 'divide and conquer' approach of its algorithm.

There are several variations of the FFT algorithm, each with their own merits. For high throughput, hardware implemented solutions, a variant of the Radix-2 Decimation-in-Time algorithm is most suitable. The 'Constant Geometry' algorithm (Fig. 1) is easier to implement whereas the 'In-Place' algorithm (Fig. 2) halves the amount of memory required.

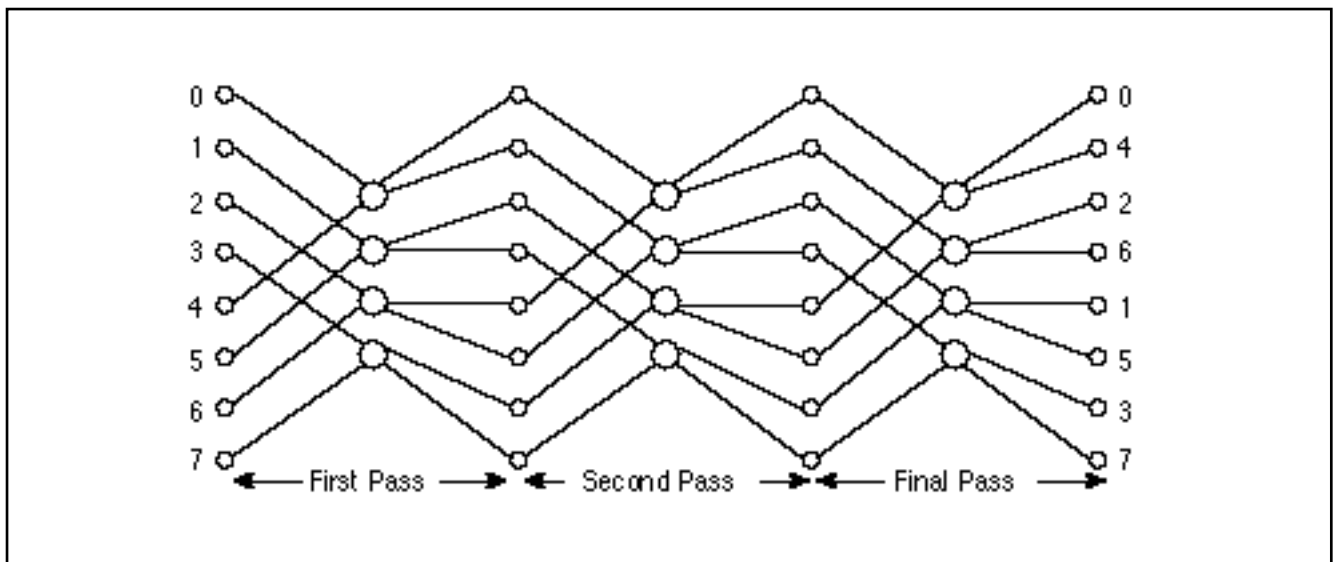


Fig.1 8 point constant geometry DIT radix 2 algorithm with normally ordered inputs and bit-reversed outputs

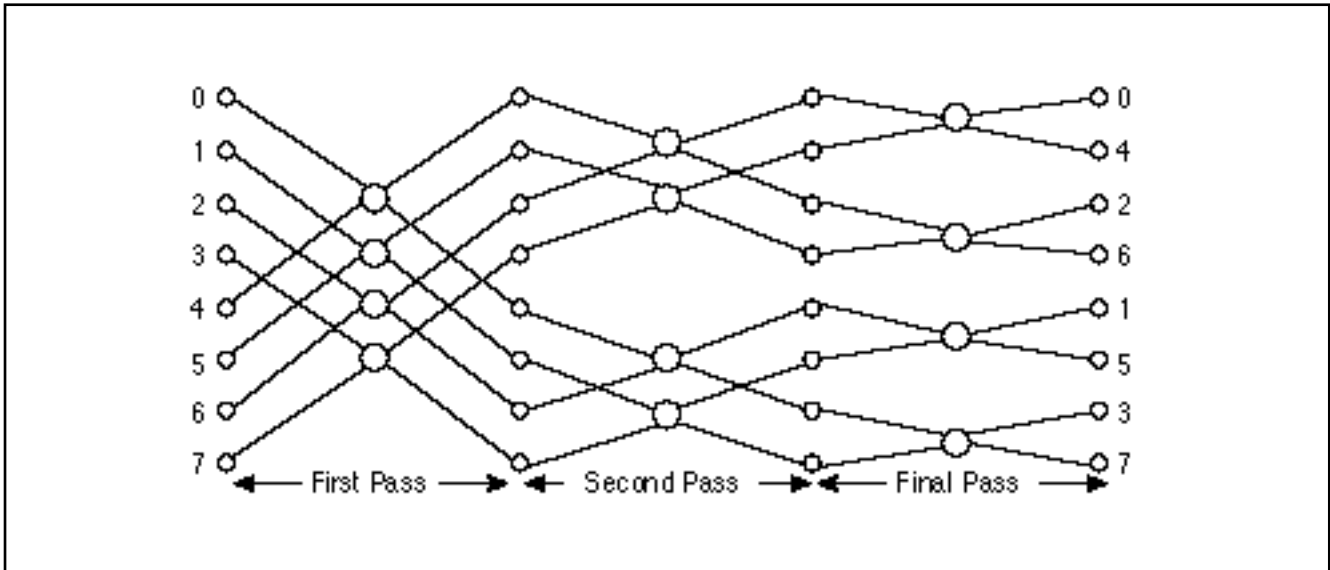
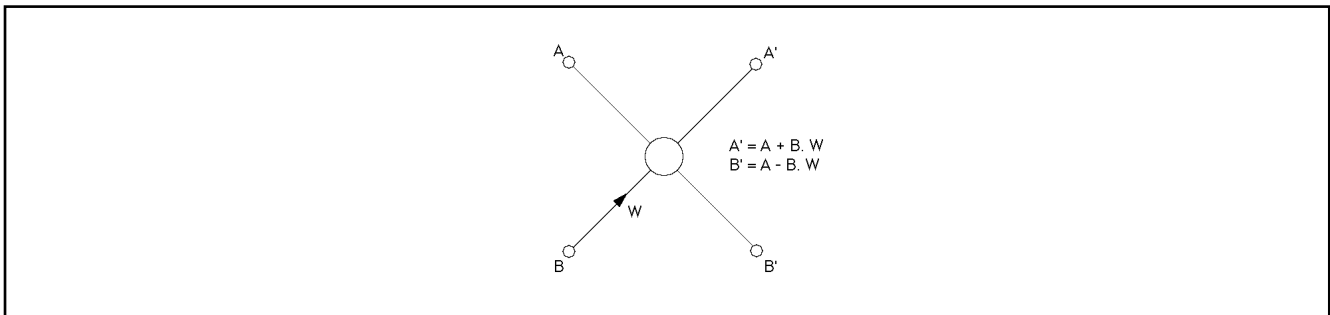


Fig.2 8 point in-place DIT radix 2 algorithm with normally ordered inputs and bit-reversed outputs

Both these variations are split vertically into a number of 'passes' ($\log_2 N$ passes for an N-point transform), each pass consisting of $N/2$ 'butterfly' operations:

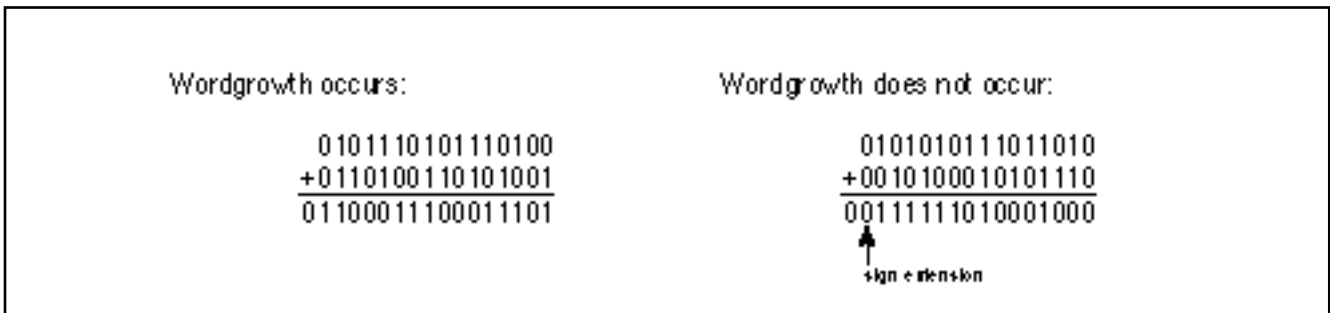


W is the complex coefficient and A and B are, for the first pass, the sampled data and then, in the second and subsequent passes, the values of A' and B' from the previous pass. The results of the FFT are the values of A' and B' from the butterflies of the final pass. In order to be compatible with previous FFT results, all points must be normalised to a universal format. These final complex number values (cartesian co-ordinates) may then be converted into magnitude and phase components (polar co-ordinates).

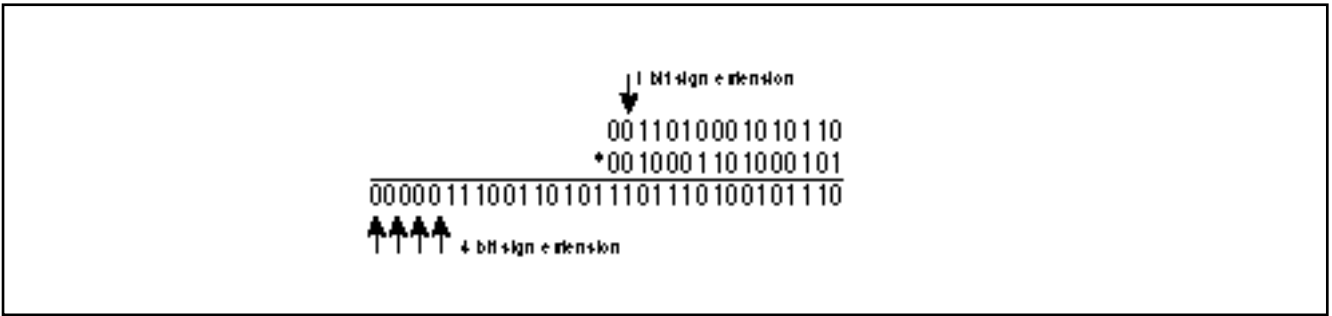
Defeating the Wordgrowth Problem

One of the most difficult problems to overcome when implementing an FFT algorithm in fixed point arithmetic is that of wordgrowth. The power of the PDSP16116s BFP system lies in its flexible and effective response to this problem. Before looking into the operation of this BFP system, the wordgrowth problem and some of the other solutions available are explained.

FFTs are implemented by means of successive multiplications and additions. Each time data is processed by an ALU (i.e. twice in each butterfly) there is the possibility of wordgrowth occurring: i.e. when two 16 bit words are added, they may produce a sum of 17 bits. The safe way to deal with this is to always pick the 16MSBs of the result. However, this will cause sign extension, i.e. repetition of the sign bit in the MSBs of the data. These two cases are illustrated in the examples below.



Sign extension can cause severe problems when the next multiplication occurs, as it is likely to lead to a product with a further extended sign bit. For example:



After a few passes of the FFT, there is a danger that the data could become all sign bits and no information - not much use to anyone. The common alternative to this approach is to pick the 16 LSBs from the ALUs and hope that no wordgrowth occurs, as this will then lead to overflow. If overflow is flagged during the course of an FFT, then the calculation must be aborted. The input data is then scaled down and the calculation repeated. The hit and miss nature of this approach can be avoided by automatically scaling down the inputs and accepting the resulting penalty in accuracy. A 'conditional shift' system offers some degree of flexibility. Here, the 16 LSBs are selected from the second ALU in the butterfly hardware if no overflow occurs in any butterfly during that pass.

The PDSP16116 offers a superior solution to the problem by employing an intelligent control system which can monitor data magnitudes during the course of the FFT and adjust them as necessary so as to keep extended sign bits to a minimum, whilst eliminating the possibility of overflow. In fact, this system can not only deal with wordgrowth problems as they occur, but can also adjust underscaled input data in anticipation of these problems to ensure that a valid result is obtained at the end of the calculation.

A comparison of the data formats provided by each of the methods detailed above will clarify their differences. Given input data of the format:

X.XXX... (note the position of the binary point)

The UNCONDITIONAL SHIFT implementation will output all data at the end of a pass in the format:

XXX.X...

regardless of whether the data has increased in magnitude or not.

The CONDITIONAL SHIFT implementation will either output ALL data in the format:

XX.XX...

if the maximum wordgrowth was one bit in any butterfly; or, if two bits of wordgrowth occurred in any butterfly, then ALL data will be output in the format:

XXX.X...

The BFP implementation can output EACH butterfly result in ANY of the following formats, according to the data magnitude:

- If data is underscaled .XXXX...
- If no wordgrowth occurs X.XXX...
- If wordgrowth occurs once XX.XX...
- If wordgrowth occurs twice XXX.X...

The adaptability of the BFP system is clearly illustrated and it is this adaptability which allows the BFP system to defeat the wordgrowth problem.

AN59

How the BFP System Operates

A block floating point system is essentially an ordinary integer arithmetic system with some additional logic, the object of which is to lend the system some of the enormous dynamic range afforded by a true floating point system without suffering the corresponding loss in performance.

The initial data used by the FFT should all have the same binary weighting, i.e. the binary point should occupy the same position in every data word. This is normal in integer arithmetic. However, during the course of the FFT, a variety of weightings are used in the data words to increase the dynamic range available. This situation is similar to that within a true floating point system, though the range of numbers representable is more limited.

In the BFP system used in the PSDP16116, there are, within any one pass of the FFT, four possible positions of the binary point within the integer words. To record the position of its binary point, each word has a 2-bit word tag associated with it. By way of example, in a particular pass we may have the following four positions of binary point available, each denoted by a certain value of word tag:

```

XX.XXXXXXXXXXXXXX      word tag = 00
XXX.XXXXXXXXXXXXXX     word tag = 01
XXXX.XXXXXXXXXXXXXX    word tag = 10
XXXXX.XXXXXXXXXXXXXX   word tag = 11
    
```

At the end of each constituent pass of the FFT, the positions of the binary point supported may change to reflect the trend of data increases or decreases in magnitude. Hence, in the pass following that of the above example, the four positions of binary point supported may change to:

```

XXXX.XXXXXXXXXXXXXX    word tag = 00
XXXXX.XXXXXXXXXXXXXX   word tag = 01
XXXXXX.XXXXXXXXXXXXXX  word tag = 10
XXXXXXX.XXXXXXXXXXXXXX word tag = 11
    
```

This variation in the range of binary points supported from pass to pass (i.e. the movement of the binary point relative to its position in the original data) is recorded in the Global Weighting Register (GWR). At the end of the final pass, the distance that the binary point has moved since the start of the FFT can be obtained by modifying the GWR according to the value of WTOUT of a particular word, as shown below:

WTOUT1:0	ADJUSTMENT TO GWR
00	SUBTRACT 1
01	NO ADJUSTMENT
10	ADD 1
11	ADD 2

For example, if the original data format was:

```
X.XXXXXXXXXXXXXX
```

then, if the GWR = 01001 and with WTOUT = 10 for a particular word, the binary point has moved 10 places to the right of its original position and will be situated as shown below:

```
XXXXXXXXXXXXX.XXXXX
```

Using the GWR with Large FFTs

The Global Weighting Register represents the movement of the binary point in two's complement notation in a 5-bit field. An examination of FFT theory and the operation of the BFP system shows that, for an N-point transform, GWR will not exceed $(2 = \log_2 N)$. This means that GWR can handle transforms as large as 8K by representing the movement of the binary point as a two's complement number. However, GWR can be used for much larger transforms by noting that GWR will never drop below -8, since with this degree of left shift, the rounding noise is amplified to fill the whole 16-bit data word. This fact allows GWR to be extended and represented as a six bit value simply by ANDing the two most significant bits to produce a new sign bit (Fig 3). This 6-bit field allows GWR to handle up to a 2097K transform.

Value of GWR	Decimal Equiv.	Meaning
00000	0	Binary point as not moved
00001	+1	Binary point has moved 1 place to the right
00010	+2	2
00011	+3	3
00100	+4	4
00101	+5	5
00110	+6	6
00111	+7	7
01000	+8	8
01001	+9	9
01010	+10	10
01011	+11	11
01100	+12	12
01101	+13	13
01110	+14	14
01111	+15	15
10000*	+16	16
10001*	+17	17
10010*	+18	18
10011*	+19	19
10100*	+20	20
10101*	+21	21
10110*	+22	22
10111*	+23	23
11000	-8	Binary point has moved 8 places to the left
11001	-7	7
11010	-6	6
11011	-5	5
11100	-4	4
11101	-3	3
11110	-2	2
11111	-1	1

* not in two's complement format

Table.1 GWR values and meanings

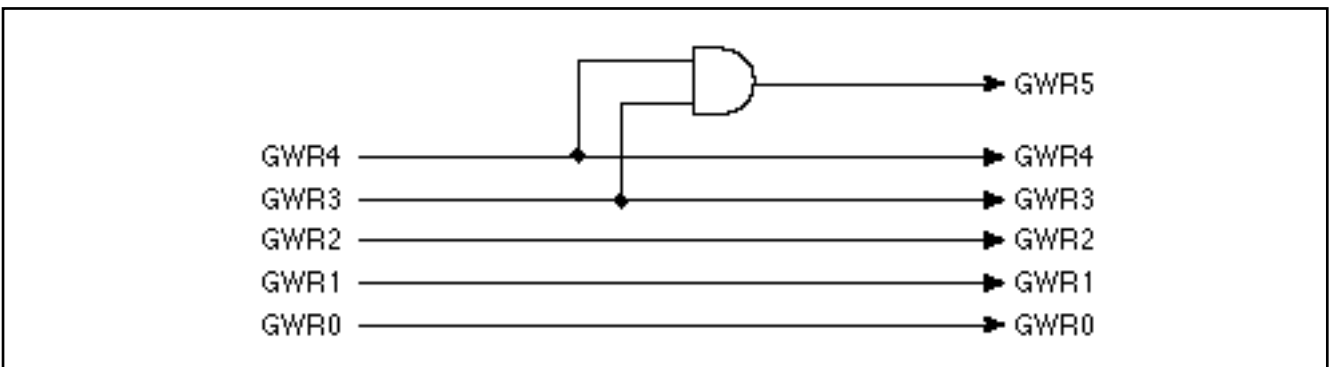


Fig.3 Extending GWR to 6 bits

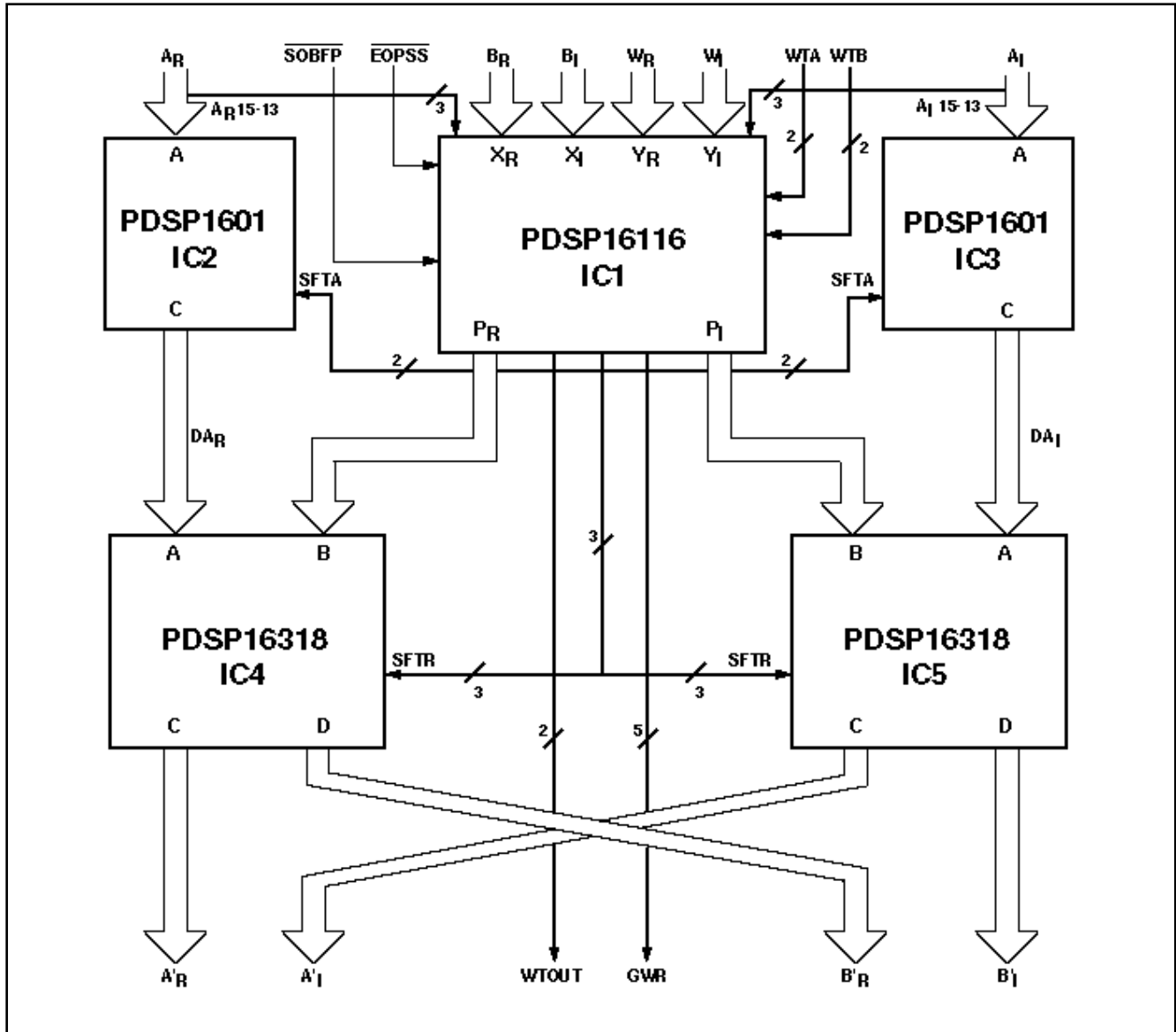


Fig.4 Block Floating Point FFT butterfly

Construction of an FFT Butterfly Processor

As described earlier, the calculations $A' = A + BW$ and $B' = A - BW$, forming a 'butterfly operation' must be carried out repeatedly in the course of an FFT. Fig. 4 shows how a butterfly processor may be constructed using a single PDSP16116 in combination with two Mitel Semiconductor's PDSP16318s and two Mitel Semiconductor's PSDP1601s. The PDSP1601s are used to match the pipeline delay and shifting operations of the PDSP16116 to the datapath of the A word. The PDSP16318s are used to perform the complex addition and subtraction of the butterfly operation. Fig. 5 details the underlying architecture of the processor.

A detailed list of the various connections required to combine these five chips into a butterfly processor appears in the Appendix. I/O connections are not specified as there are a number of I/O options that allow the butterfly processor to be interfaced with the rest of an FFT system.

A point to note is the hard-wired 1-bit right shift in the A-word data paths between the PDSP16116 outputs and the PDSP16318 inputs. This is to keep the A-word data format the same as the PDSP16116 output data format so that the two words may be added within the PDSP16318. The PDSP1601 applies a shift of 0 to 3 places to the right whereas data is output from the PDSP16116 with the binary point shifted from 1 to 4 places to the right. Hence an extra right shift of one place needs to be inserted in the PDSP1601 data path to keep the data formats compatible at the inputs to the PDSP16318 (data words must have their binary points in the same places before being added).

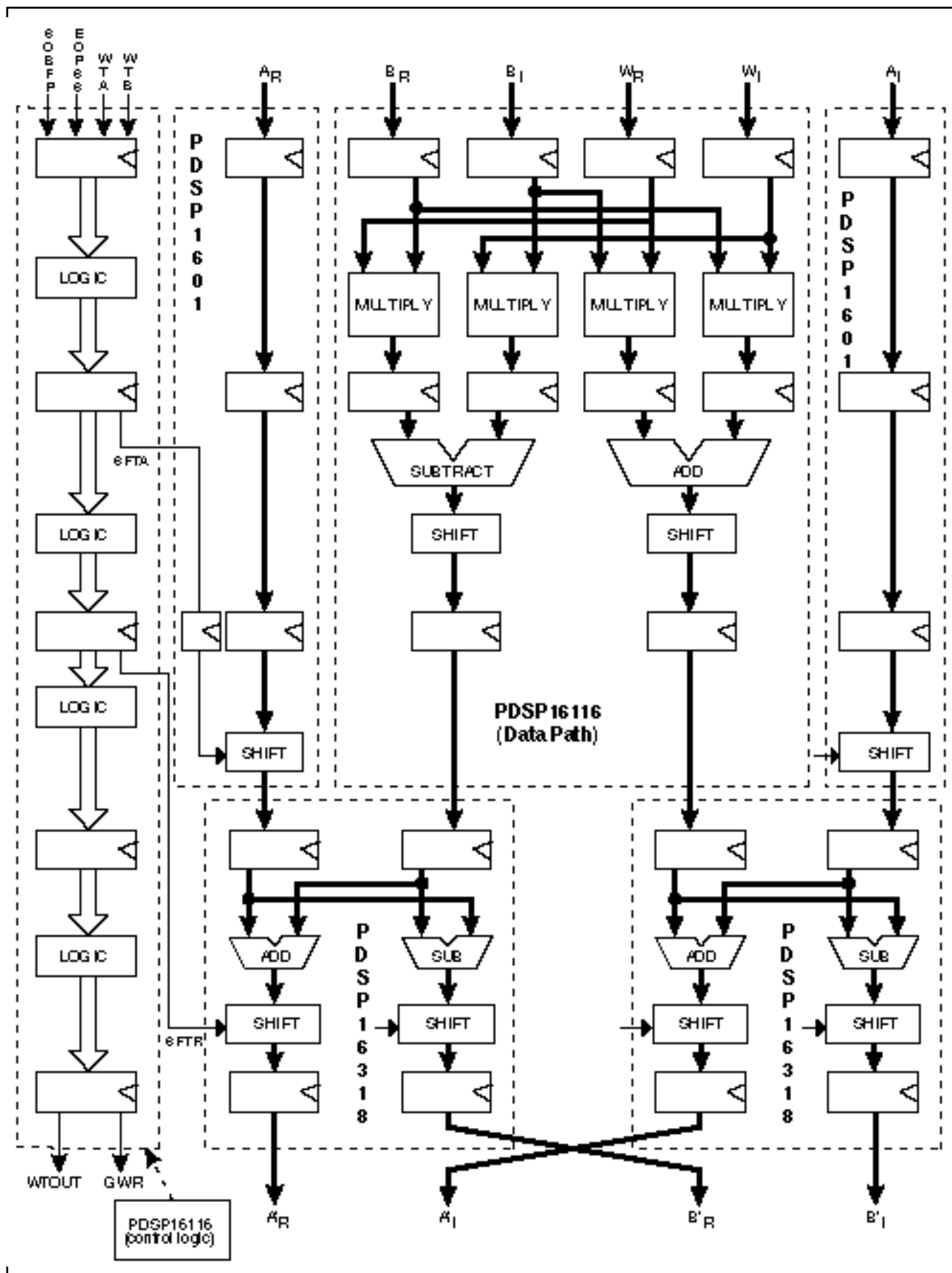


Fig.5 BFP Butterfly Detail

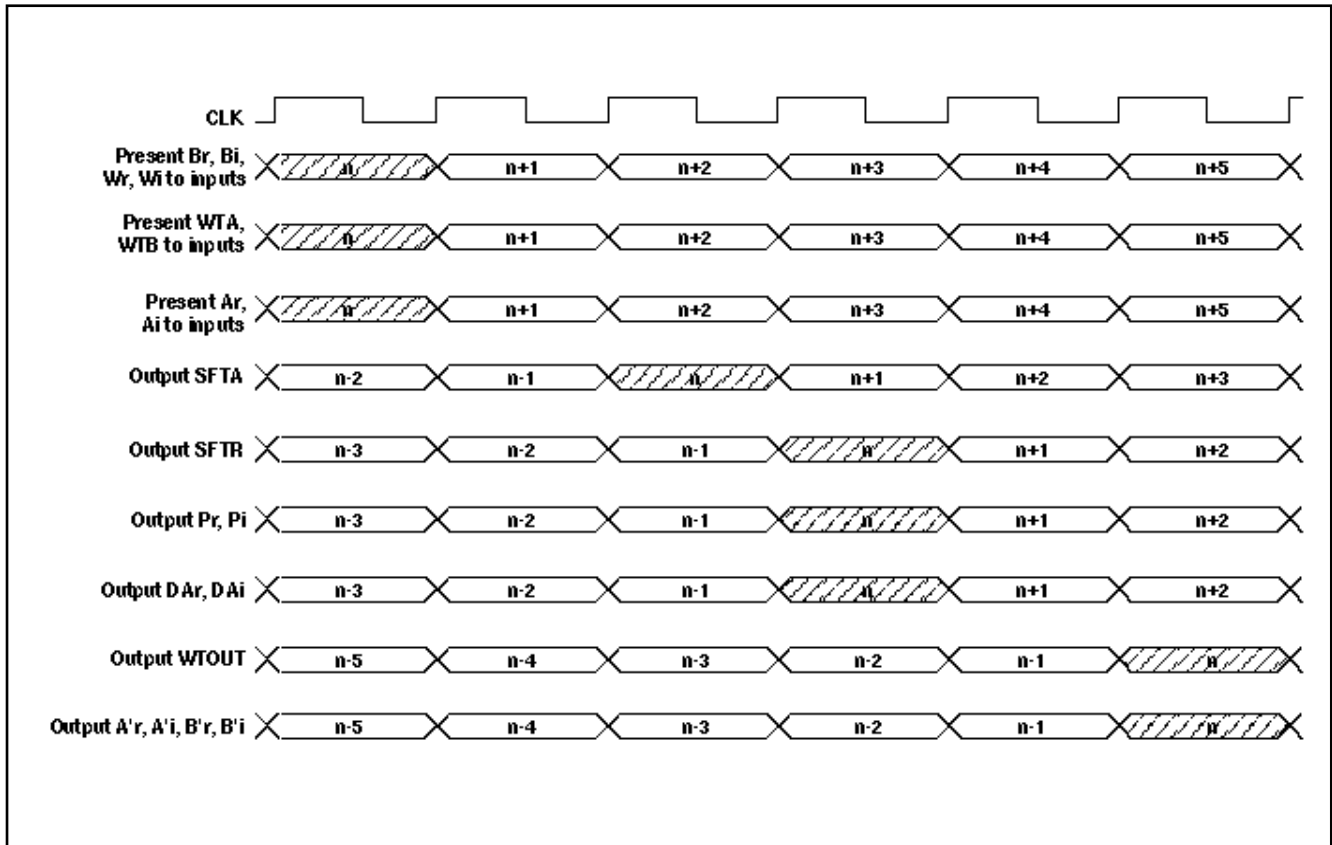


Fig.6 Data and Control Timing in the Butterfly

The Butterfly Operation

A new butterfly operation is commenced each cycle, requiring a new set of data for A, B, W, WTA and WTB. Five cycles later, the corresponding results A' and B' are produced along with their associated WTOUT. In between, the signals SFTA and SFTR are produced and acted upon by the shifters in the PDSP1601 and PDSP16318. The timing of the data and control signals is shown in Fig. 6.

The results (A' and B') of each butterfly calculation in a pass must be stored away to be used later as the input data (A and B) in the next pass. In every pass, each result must be stored together with its associated word tag, WTOUT. Although WTOUT is common to both A' and B', it must be stored separately with each word as the words are used on different cycles during the next pass. At the inputs, the word tag associated with the A word is known as WTA and the word tag associated with the B word is known as WTB. Hence the WTOUTs from one pass will become the WTAs and the WTBs for the following pass. It should be noted that the first pass is unique in that word tags need not be input into the butterfly as all data must initially have the same weighting. Therefore, during the first pass alone, the inputs WTA and WTB are ignored.

Control of the FFT

To enable the block floating point hardware to keep track of the data, the following signals are provided:

- SOBFP - start of the FFT
- EOPSS - end of current pass

These inform the PDSP16116 when an FFT is starting and when each pass is complete. Fig. 7 shows the timing of these signals and an explanation of their use follows.

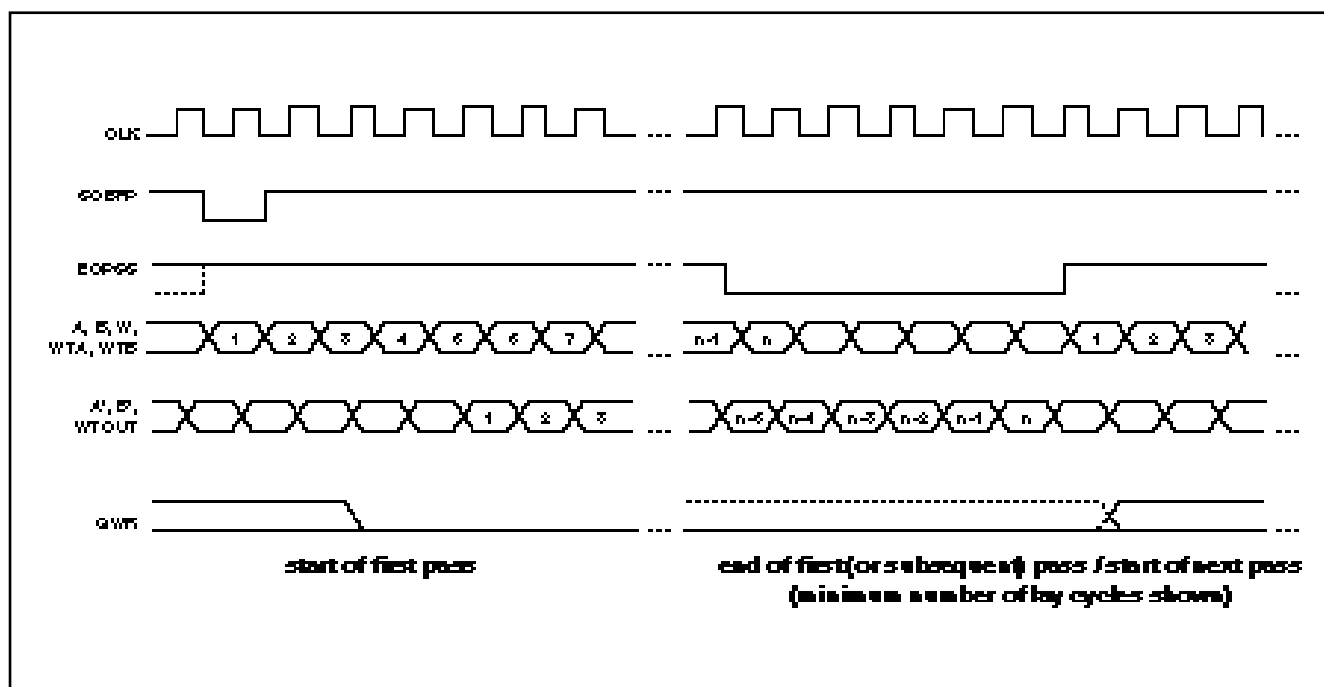


Fig.7 Use of BFP Control Signals

To commence the FFT, the signal EOPSS should be set high (where it will remain for the duration of the pass). SOBFP should be pulled low during the initial cycle, when the first data words A and B are presented to the inputs of the butterfly processor. The following cycle, SOBFP must be pulled high where it should remain for the duration of the FFT. New data is presented to the processor each successive cycle until the end of the first pass of the FFT. On the last cycle of the pass, the signal EOPSS should be pulled low and remain low for a minimum of five cycles, the time required to clear the pipeline of the butterfly processor so that all the results from one pass are obtained before commencing the following pass (should a longer pause be required between passes - to arrange the data for the next pass, for example - then EOPSS may be kept low for as long as necessary, the next pass cannot commence until it is brought high again). On the initial cycle of each new pass, the signal EOPSS should be pulled high and it should remain high until the final cycle of that pass, when it is pulled low again.

Building an FFT System

The Butterfly Processor is only one element of a complete FFT system. Also required are fast A/D converters at the front end of the system; a complex heterodyne filter to zoom-in on the frequencies of interest; fast memory and addressing circuits to store the data; additional fast memory and addressing circuits for the coefficients; an output normalisation circuit to make all data consistent; a Pythagoras Processor to extract magnitude and phase information from the results; finally, a D/A converter to allow the magnitude and phase information to be displayed on a video screen or oscilloscope. Fig. 8 shows how these blocks are connected. Mitel Semiconductor makes a range of high performance DSP devices which solve the more difficult problems outlined above. The complex heterodyne filter may be constructed from a combination of either a PDSP16116 or PDSP16112 complex multiplier and either a PDSP16318 complex accumulator or two PDSP1601 augmented arithmetic logic units. Output normalisation is a simple matter with the PDSP1601's adaptable barrel shifter and the PDSP16330 Pythagoras processor to convert Cartesian to polar coordinates.

Memory Requirements

Memory requirements differ according to whether the 'In-Place' or 'Constant Geometry' algorithms are used. In either case, two reads from memory (A and B) and two writes to memory (A' and B') have to be made each 100ns cycle.

For the In-Place algorithm, the results (A', B') of a butterfly are written to the same locations from which the inputs (A & B) were read. Hence, the memory must have an access time of 25ns to cope with the two reads and two writes.

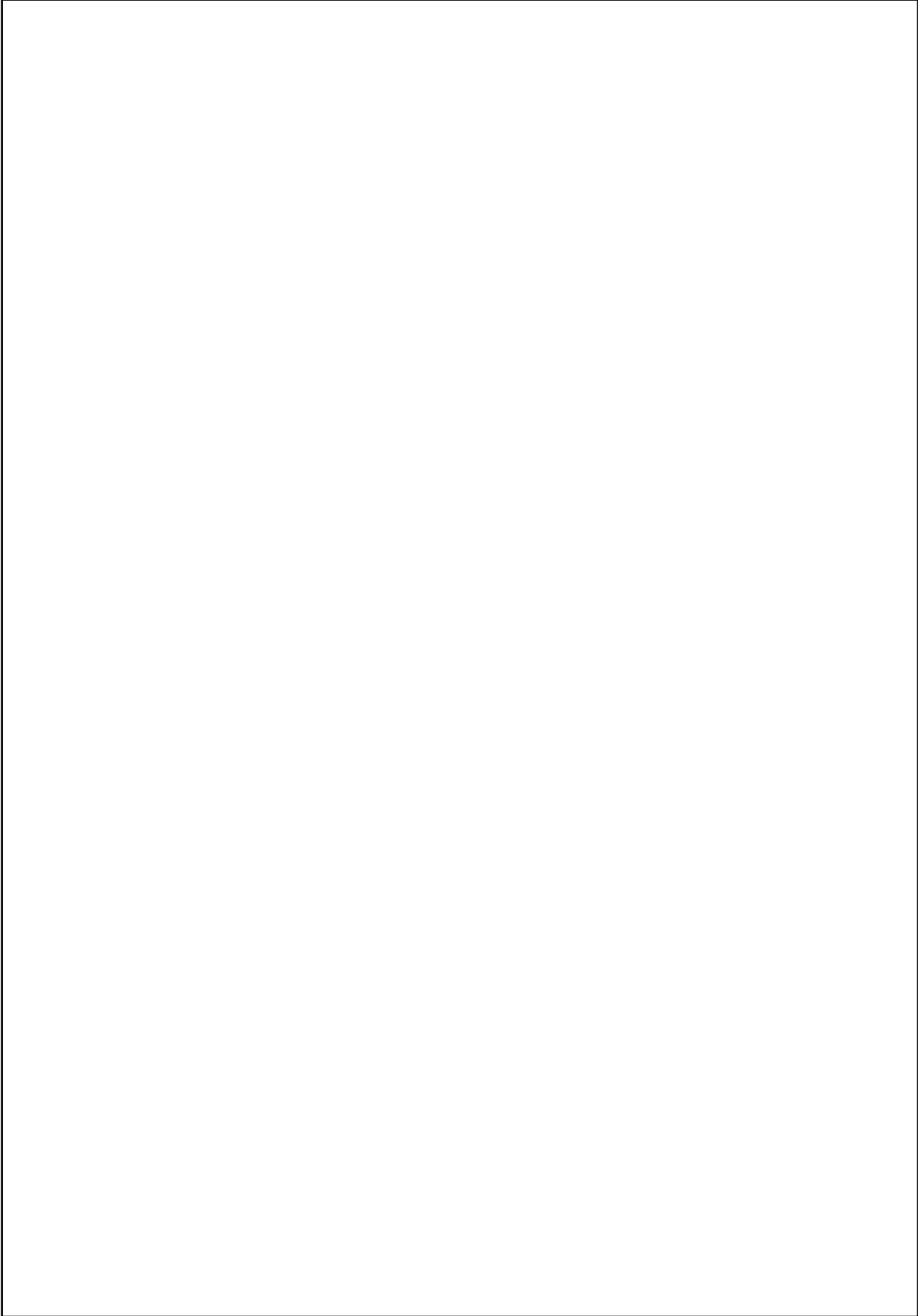


Fig.8 Typical FFT System

The Constant Geometry algorithm requires a memory access time of only 50ns, but the memory size must be double that of the In-Place algorithm. This is because the addresses written to after each butterfly are different from those from which the input data was read. This is possible due to the order in which data points are addressed.

The memory must be 32 bits wide to accommodate the real and imaginary parts of each word. Also, the 2 bit word tag must be stored with each word. This could be achieved by widening the memory to 34 bits or, alternatively, it could be stored in the LSB of the real and imaginary parts of the word, keeping the memory width at 32 bits. This would not affect the accuracy of the FFT, as the LSB is a rounded value in any case. There would be no problem in the initial pass when no word tags have been written to the memory as the PDSP16116 ignores the word tag inputs during the initial pass.

FFT Output Normalisation

In order to preserve the dynamic range of the data during the FFT calculation, the PDSP16116 employs a range of different weightings, however, at the end of the FFT, the data must be re-formatted to a pre-determined common weighting. This can be done by comparing the exponent of a given data word with the required universal exponent and then shifting the data word by the difference. The PDSP1601 ALU, with its multifunction 16-bit barrel shifter, is ideally suited to this task.

What value should the universal exponent take ? Theoretically, the largest possible data result from an FFT is $1.27N$ times the largest input data, where N is the size of the FFT. This means that the binary point can move a maximum of $(1 + \log_2 N)$ places to the right. Hence, if the universal exponent is chosen to be $(1 + \log_2 N)$, this should give a sufficient range to represent all data points faithfully. In practice, the FFT output data may never approach the theoretical maximum, therefore it may be worthwhile trying various universal exponents and choosing the one best suited to the particular application.

Data is output from the butterfly processor with a two part exponent: the 5-bit GWR applicable to all data words from a given FFT and a 2-bit WTOUT associated with each individual data word. To find the complete exponent for a given word, the GWR for that FFT must be modified by the WTOUT value, the result being the number of places that the binary point has been shifted to the right during the course of the FFT. This value must be subtracted from the universal exponent, the difference being the shift required for that data word, which is input to the SV port of the PDSP1601.

As FFT data consists of real and imaginary parts, either two PDSP1601s must be used or a single PDSP1601 handling real and imaginary data on alternate cycles, the same shift being applied to both parts. An example of an output normalisation circuit is shown in Fig. 9. Only 4-bit arithmetic is used in calculating the shift which means that very small (negative) values of GWR must be trapped and a forced 16-bit right shift applied. (NB. It is easier to simply add the word tag value to the GWR to determine the shift rather than modifying it exactly. To compensate for this, the universal exponent should be increased by one.)

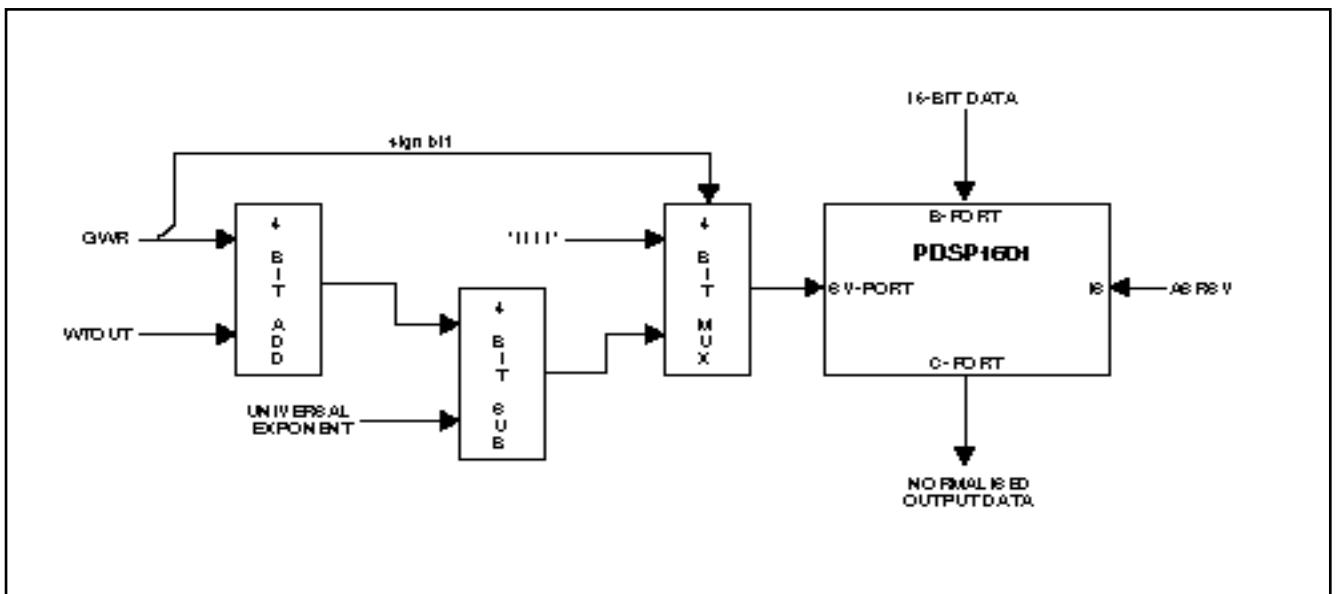


Fig.9 Output Normalisation Circuitry

AN59

Appendix A - Block Floating Point FFT Butterfly Net List

The following net lists give all the connections required for implementing the Block Floating Point FFT butterfly shown in Fig.4 :

IC1: PDSP16116 Complex Multiplier

Pin No.	Pin desc.	Net Name	Connections
D3	PI14	PI14	IC5-C11
C2	PI15	PI15	IC5-D10
B1	WTOUT1	WTOUT1	external o/p
D2	WTOUT0	WTOUT0	external o/p
E3	SFTR0	SFTR0	IC4-L7; IC5-L7
C1	SFTR1	SFTR1	IC4-J7; IC5-J7
E2	SFTR2	SFTR2	IC4-J6; IC5-J6
D1	OEI		tie low
F3	CONX		tie low
F2	CONY		tie low
E1	ROUND		tie high
G2	AI13	AI13	external i/p; IC3-H1
G3	AI14	AI14	external i/p; IC3-F1
F1	AI15	AI15	external i/p; IC3-G2
G1	AR13	AR13	external i/p; IC2-H1
H2	AR14	AR14	external i/p; IC2-F1
H1	AR15	AR15	external i/p; IC2-G2
H3	WI15	WI15	external i/p
J3	WI14	WI14	external i/p
J1	WI13	WI13	external i/p
K1	WI12	WI12	external i/p
J2	WI11	WI11	external i/p
K2	WI10	WI10	external i/p
K3	WI9	WI9	external i/p
L1	WI8	WI8	external i/p
L2	WI7	WI7	external i/p
M1	WI6	WI6	external i/p
N1	WI5	WI5	external i/p
M2	WI4	WI4	external i/p
L3	WI3	WI3	external i/p
N2	WI2	WI2	external i/p
P1	WI1	WI1	external i/p
M3	WI0	WI0	external i/p
N3	BI0	BI0	external i/p
P2	GND	GND	0V supply rail
R1	VDD	VDD	+5V supply rail
N4	BI1	BI1	external i/p
P3	BI2	BI2	external i/p
R2	BI3	BI3	external i/p
P4	BI4	BI4	external i/p
N5	BI5	BI5	external i/p
R3	BI6	BI6	external i/p
P5	BI7	BI7	external i/p
R4	BI8	BI8	external i/p
N6	BI9	BI9	external i/p
P6	BI10	BI10	external i/p
R5	BI11	BI11	external i/p
P7	BI12	BI12	external i/p
N7	BI13	BI13	external i/p
R6	BI14	BI14	external i/p
R7	BI15	BI15	external i/p

IC1: PDSP16116 Complex Multiplier (Continued)

Pin No.	Pin desc.	Net Name	Connections
P8	CEY		tie low
R8	CEX		tie low
N8	XR15	BR15	external i/p
N9	XR14	BR14	external i/p
R9	XR13	BR13	external i/p
R10	XR12	BR12	external i/p
P9	XR11	BR11	external i/p
P10	XR10	BR10	external i/p
N10	XR9	BR9	external i/p
R11	XR8	BR8	external i/p
P11	XR7	BR7	external i/p
R12	XR6	BR6	external i/p
R13	XR5	BR5	external i/p
P12	XR4	BR4	external i/p
N11	XR3	BR3	external i/p
P13	XR2	BR2	external i/p
R14	XR1	BR1	external i/p
N12	XR0	BR0	external i/p
N13	YR15	WR15	external i/p
P14	YR14	WR14	external i/p
R15	YR13	WR13	external i/p
M13	GND	GND	0V supply rail
N14	VDD	VDD	+5V supply rail
P15	YR12	WR12	external i/p
M14	YR11	WR11	external i/p
L13	YR10	WR10	external i/p
N15	YR9	WR9	external i/p
L14	YR8	WR8	external i/p
M15	YR7	WR7	external i/p
K13	YR6	WR6	external i/p
K14	YR5	WR5	external i/p
L15	YR4	WR4	external i/p
J14	YR3	WR3	external i/p
J13	YR2	WR2	external i/p
K15	YR1	WR1	external i/p
J15	YR0	WR0	external i/p
H14	EOPSS	EOPSS	external i/p
H15	VDD	VDD	+5V supply rail
H13	SOBFP	SOBFP	external i/p
G13	WTB1	WTB1	external i/p
G15	WTB0	WTB0	external i/p
F15	WTA1	WTA1	external i/p
G14	WTA0	WTA0	external i/p
F14	MBFP		tie high
F13	CLK	CLK	external i/p - common to all ICs
E15	OSEL1		tie low
E14	OSEL0		tie low
D15	OER		tie low
C15	SFTA0	SFTA0	IC2-L6; IC3-L6
D14	SFTA1	SFTA1	IC2-L8; IC3-L8
E13	GWR0	GWR0	external o/p
C14	GWR1	GWR1	external o/p
B15	GWR2	GWR2	external o/p
D13	GWR3	GWR3	external o/p
C13	GWR4	GWR4	external io/p
B14	PR15	PR15	IC4-D10
A15	PR14	PR14	IC4-C11

AN59

IC1: PDSP16116 Complex Multiplier (Continued)

Pin No.	Pin desc.	Net Name	Connections
C12	VDD	VDD	+5V supply rail
B13	GND	GND	0V supply rail
A14	PR13	PR13	IC4-B11
B12	PR12	PR12	IC4-C10
C11	PR11	PR11	IC4-A11
A13	PR10	PR10	IC4-B10
B11	PR9	PR9	IC4-B9
A12	PR8	PR8	IC4-A10
C10	PR7	PR7	IC4-A9
B10	PR6	PR6	IC4-B8
A11	PR5	PR5	IC4-A8
B9	GND	GND	0V supply rail
C9	VDD	VDD	+5V supply rail
A10	PR4	PR4	IC4-B6
A9	PR3	PR3	IC4-B7
B8	PR2	PR2	IC4-A7
A8	PR1	PR1	IC4-C7
C8	PR0	PR0	IC4-C6
C7	PI0	PI0	IC5-C6
A7	PI1	PI1	IC5-C7
A6	PI2	PI2	IC5-A7
B7	PI3	PI3	IC5-B7
B6	PI4	PI4	IC5-B6
C6	VDD	VDD	+5V supply rail
A5	PI5	PI5	IC5-A8
B5	GND	GND	0V supply rail
A4	PI6	PI6	IC5-B8
A3	PI7	PI7	IC5-A9
B4	PI8	PI8	IC5-A10
C5	PI9	PI9	IC5-B9
B3	PI10	PI10	IC5-B10
A2	PI11	PI11	IC5-A11
C4	PI12	PI12	IC5-C10
C3	PI13	PI13	IC5-B11
B2	GND	GND	0V supply rail
A1	VDD	VDD	+5V supply rail

IC2: PDSP1601 - Real Path

Pin No.	Pin desc.	Net Name	Connections
B10	VCC	VDD	+5V supply rail
A6	MSB		tie low
A5	MSS		tie high
B5	B15		tie low
C5	B14		tie low
A4	B13		tie low
B4	B12		tie low
A3	B11		tie low
A2	B10		tie low
B3	B9		tie low
A1	B8		tie low
B2	B7		tie low
C2	B6		tie low
B1	B5		tie low
C1	B4		tie low
D2	B3		tie low
D1	B2		tie low
E3	B1		tie low
E2	B0		tie low
E1	CEB		tie high
F2	CLK	CLK	external i/p - common to all ICs
F3	GND	GND	0V supply rail
G3	MSA0		tie high
G1	MSA1		tie low
G2	A15	AR15	external i/p ; IC1-H1
F1	A14	AR14	external i/p ; IC1 - H2
H1	A13	AR13	external i/p ; IC1-G1
H2	A12	AR12	external i/p
J1	A11	AR11	external i/p
K1	A10	AR10	external i/p
J2	A9	AR9	external i/p
L1	A8	AR8	external i/p
K2	A7	AR7	external i/p
K3	A6	AR6	external i/p
L2	A5	AR5	external i/p
L3	A4	AR4	external i/p
K4	A3	AR3	external i/p
L4	A2	AR2	external i/p
J5	A1	AR1	external i/p
K5	A0	AR0	external i/p
L5	CEA		tie low
K6	MSC		tie high
K10	VCC	VDD	+5V supply rail
J6	IS0		tie low
J7	IS1		tie high
L7	IS2		tie low
K7	IS3		tie high
L6	SV0	SFTA0	IC1-C15
L8	SV1	SFTA1	IC1-D14
K8	SV2		tie low
L9	SV3		tie low
L10	SVOE		tie high
K9	RS0		tie high
L11	RS1		tie high
J10	RS2		tie high

AN59

IC2: PDSP1601 - Real Path (continued)

Pin No.	Pin desc.	Net Name	Connections
K11	C0		N/C
J11	C1	DAR0	IC4-L11
H10	C2	DAR1	IC4-K10
H11	C3	DAR2	IC4-J10
F10	C4	DAR3	IC4-K11
G10	C5	DAR4	IC4-J11
G11	C6	DAR5	IC4-H10
G9	C7	DAR6	IC4-H11
F9	GND	GND	0V supply rail
F11	C8	DAR7	IC4-F10
E11	C9	DAR8	IC4-G10
E10	C10	DAR9	IC4-G11
E9	C11	DAR10	IC4-G9
D11	C12	DAR11	IC4-F9
D10	C13	DAR12	IC4-F11
C11	C14	DAR13	IC4-E11
B11	C15	DAR14:15	IC4-E9, E10
C10	OE		tie low
A11	BFP		N/C
B9	CO		N/C
A10	RA0		L on even cycles, H on odd cycles
A9	RA1		tie high
B8	RA2		tie low
A8	CI		tie low
B6	IA0		tie low
B7	IA1		tie high
A7	IA2		tie high
C7	IA3		tie low
C6	IA4		tie high

IC3: PDSP1601 - Imaginary Path

Pin No.	Pin desc.	Net Name	Connections
B10	VCC	VDD	+5V supply rail
A6	MSB		tie low
A5	MSS		tie high
B5	B15		tie low
C5	B14		tie low
A4	B13		tie low
B4	B12		tie low
A3	B11		tie low
A2	B10		tie low
B3	B9		tie low
A1	B8		tie low
B2	B7		tie low
C2	B6		tie low
B1	B5		tie low
C1	B4		tie low
D2	B3		tie low
D1	B2		tie low
E3	B1		tie low
E2	B0		tie low
E1	CEB		tie high
F2	CLK	CLK	external i/p - common to all ICs
F3	GND	GND	0V supply rail
G3	MSA0		tie high
G1	MSA1		tie low
G2	A15	AI15	external i/p ; IC1-F1
F1	A14	AI14	external i/p ; IC1 - G3
H1	A13	AI13	external i/p ; IC1-G2
H2	A12	AI12	external i/p
J1	A11	AI11	external i/p
K1	A10	AI10	external i/p
J2	A9	AI9	external i/p
L1	A8	AI8	external i/p
K2	A7	AI7	external i/p
K3	A6	AI6	external i/p
L2	A5	AI5	external i/p
L3	A4	AI4	external i/p
K4	A3	AI3	external i/p
L4	A2	AI2	external i/p
J5	A1	AI1	external i/p
K5	A0	AI0	external i/p
L5	CEA		tie low
K6	MSC		tie high
K10	VCC	VDD	+5V supply rail
J6	IS0		tie low
J7	IS1		tie high
L7	IS2		tie low
K7	IS3		tie high
L6	SV0	SFTA0	IC1-C15
L8	SV1	SFTA1	IC1-D14
K8	SV2		tie low
L9	SV3		tie low
L10	SVOE		tie high
K9	RS0		tie high
L11	RS1		tie high
J10	RS2		tie high

AN59

IC3: PDSP1601 - Imaginary Path (continued)

Pin No.	Pin desc.	Net Name	Connections
K11	C0		N/C
J11	C1	DAI0	IC5-L11
H10	C2	DAI1	IC5-K10
H11	C3	DAI2	IC5-J10
F10	C4	DAI3	IC5-K11
G10	C5	DAI4	IC5-J11
G11	C6	DAI5	IC5-H10
G9	C7	DAI6	IC5-H11
F9	GND	GND	0V supply rail
F11	C8	DAI7	IC5-F10
E11	C9	DAI8	IC5-G10
E10	C10	DAI9	IC5-G11
E9	C11	DAI10	IC5-G9
D11	C12	DAI11	IC5-F9
D10	C13	DAI12	IC5-F11
C11	C14	DAI13	IC5-E11
B11	C15	DAI14:15	IC5-E9, E10
C10	OE		tie low
A11	BFP		N/C
B9	CO		N/C
A10	RA0		L on even cycles, H on odd cycles
A9	RA1		tie high
B8	RA2		tie low
A8	CI		tie low
B6	IA0		tie low
B7	IA1		tie high
A7	IA2		tie high
C7	IA3		tie low
C6	IA4		tie high

IC4: PDSP16318 - Real Path

Pin No.	Pin desc.	Net Name	Connections
B2	D7	B'R7	external o/p
C2	D8	B'R8	external o/p
B1	D9	B'R9	external o/p
C1	D10	B'R10	external o/p
D2	GND	GND	0V supply rail
D1	VDD	VDD	+5V supply rail
E3	D11	B'R11	external o/p
E2	D12	B'R12	external o/p
E1	D13	B'R13	external o/p
F2	D14	B'R14	external o/p
F3	D15	B'R15	external o/p
G3	C15	A'R15	external o/p
G1	C14	A'R14	external o/p
G2	C13	A'R13	external o/p
F1	C12	A'R12	external o/p
H1	VDD	VDD	+5V supply rail
H2	GND	GND	0V supply rail
J1	C11	A'R11	external o/p
K1	C10	A'R10	external o/p
J2	C9	A'R9	external o/p
L1	C8	A'R8	external o/p
K2	C7	A'R7	external o/p
K3	C6	A'R6	external o/p
L2	C5	A'R5	external o/p
L3	C4	A'R4	external o/p
K4	C3	A'R3	external o/p
L4	C2	A'R2	external o/p
J5	C1	A'R1	external o/p
K5	C0	A'R0	external o/p
L5	OED		tie low
K6	OEC		tie low
J6	SD2	SFTR2	IC1-E2 ; IC5-J6
J7	SD1	SFTR1	IC1-C1 ; IC5-J7
L7	SD0	SFTR0	IC1-E3 ; IC5-L7
K7	MS		tie low
L6	AS11		tie high
L8	AS10		tie low
K8	DEL		tie low
L9	CLR		tie low
L10	ASR1		tie low
K9	ASR0		tie low
L11	A0	DAR0	IC2-J11
K10	A1	DAR1	IC2-H10
J10	A2	DAR2	IC2-H11
K11	A3	DAR3	IC2-F10
J11	A4	DAR4	IC2-G10
H10	A5	DAR5	IC2-G11
H11	A6	DAR6	IC2-G9
F10	A7	DAR7	IC2-F11
G10	A8	DAR8	IC2-E11
G11	A9	DAR9	IC2-E10
G9	A10	DAR10	IC2-E9
F9	A11	DAR11	IC2-D11
F11	A12	DAR12	IC2-D10
E11	A13	DAR13	IC2-C11
E10	A14	DAR14	IC2-B11
E9	A15	DAR15	IC2-B11

AN59

IC4: PDSP16318 - Real Path (continued)

Pin No.	Pin desc.	Net Name	Connections
D11	CEA		tie low
D10	B15	PR15	IC1-B14
C11	B14	PR14	IC1-A15
B11	B13	PR13	IC1-A14
C10	B12	PR12	IC1-B12
A11	B11	PR11	IC1-C11
B10	B10	PR10	IC1-A13
B9	B9	PR9	IC1-B11
A10	B8	PR8	IC1-A12
A9	B7	PR7	IC1-C10
B8	B6	PR6	IC1-B10
A8	B5	PR5	IC1-A11
B6	B4	RP4	IC1-A10
B7	B3	PR3	IC1-A9
A7	B2	PR2	IC1-B8
C7	B1	PR1	IC1-A8
C6	B0	PR0	IC1-C8
A6	CLK	CLK	external i/p - common to all ICs
A5	CEB		tie low
B5	OVR		N/C
C5	D0	B'R0	external o/p
A4	D1	B'R1	external o/p
B4	D2	B'R2	external o/p
A3	D3	B'R3	external o/p
A2	D4	B'R4	external o/p
B3	D5	B'R5	external o/p
A1	D6	B'R6	external o/p

IC5: PDSP16318 - Imaginary Path

Pin No.	Pin desc.	Net Name	Connections
B2	D7	B'17	external o/p
C2	D8	B'18	external o/p
B1	D9	B'19	external o/p
C1	D10	B'110	external o/p
D2	GND	GND	0V supply rail
D1	VDD	VDD	+5V supply rail
E3	D11	B'111	external o/p
E2	D12	B'112	external o/p
E1	D13	B'113	external o/p
F2	D14	B'114	external o/p
F3	D15	B'115	external o/p
G3	C15	A'115	external o/p
G1	C14	A'114	external o/p
G2	C13	A'113	external o/p
F1	C12	A'112	external o/p
H1	VDD	VDD	+5V supply rail
H2	GND	GND	0V supply rail
J1	C11	A'111	external o/p
K1	C10	A'110	external o/p
J2	C9	A'19	external o/p
L1	C8	A'18	external o/p
K2	C7	A'17	external o/p
K3	C6	A'16	external o/p
L2	C5	A'15	external o/p
L3	C4	A'14	external o/p
K4	C3	A'13	external o/p
L4	C2	A'12	external o/p
J5	C1	A'11	external o/p
K5	C0	A'10	external o/p
L5	OED		tie low
K6	OEC		tie low
J6	SD2	SFTR2	IC1-E2 ; IC4-J6
J7	SD1	SFTR1	IC1-C1 ; IC4-J7
L7	SD0	SFTR0	IC1-E3 ; IC4-L7
K7	MS		tie low
L6	ASI1		tie high
L8	ASI0		tie low
K8	DEL		tie low
L9	CLR		tie low
L10	ASR1		tie low
K9	ASR0		tie low
L11	A0	DAI0	IC3-J11
K10	A1	DAI1	IC3-H10
J10	A2	DAI2	IC3-H11
K11	A3	DAI3	IC3-F10
J11	A4	DAI4	IC3-G10
H10	A5	DAI5	IC3-G11
H11	A6	DAI6	IC3-G9
F10	A7	DAI7	IC3-F11
G10	A8	DAI8	IC3-E11
G11	A9	DAI9	IC3-E10
G9	A10	DAI10	IC3-E9
F9	A11	DAI11	IC3-D11
F11	A12	DAI12	IC3-D10
E11	A13	DAI13	IC3-C11
E10	A14	DAI14	IC3-B11
E9	A15	DAI15	IC3-B11

AN59

IC5: PDSP16318 - Imaginary Path (continued)

Pin No.	Pin desc.	Net Name	Connections
D11	CEA		tie low
D10	B15	PI15	IC1-C2
C11	B14	PI14	IC1-D3
B11	B13	PI13	IC1-C3
C10	B12	PI12	IC1-C4
A11	B11	PI11	IC1-A2
B10	B10	PI10	IC1-B3
B9	B9	PI9	IC1-C5
A10	B8	PI8	IC1-B4
A9	B7	PI7	IC1-A3
B8	B6	PI6	IC1-A4
A8	B5	PI5	IC1-A5
B6	B4	PI4	IC1-B6
B7	B3	PI3	IC1-B7
A7	B2	PI2	IC1-A6
C7	B1	PI1	IC1-A7
C6	B0	PI0	IC1-C7
A6	CLK	CLK	external i/p - common to all ICs
A5	CEB		tie low
B5	OVR		N/C
C5	D0	B'I0	external o/p
A4	D1	B'I1	external o/p
B4	D2	B'I2	external o/p
A3	D3	B'I3	external o/p
A2	D4	B'I4	external o/p
B3	D5	B'I5	external o/p
A1	D6	B'I6	external o/p

AN59

References

For a general introduction to FFTs the following texts are recommended:

1. Rabiner and Gold, 'Theory and Application of Digital Signal Processing', Prentice - Hall 1975
2. Oppenheim and Shafer, 'Digital Signal Processing', Prentice - Hall 1975

Other Mitel Semiconductor applications notes and briefs of interest include:

- AN47 'A Radix 2 Butterfly Processor'
- AN49 'Complex Signal Processing with the PSDP16000 Family'
- AN50 'A Fast FFT Processor using the PDSP16000 Family'
- AB01 'A 50ns Butterfly Processor'
- AB10 'FIR Filtering with the PDSP16112 and PDSP16318'

In addition, many PDSP devices and applications are modelled on the 'PDSP Demonstrator' software, intended to be run on IBM-PC or compatibles.



HEADQUARTERS OPERATIONS
MITEL SEMICONDUCTOR
Cheney Manor, Swindon,
Wiltshire SN2 2QW, United Kingdom.
Tel: (01793) 518000
Fax: (01793) 518411

MITEL SEMICONDUCTOR
1500 Green Hills Road,
Scotts Valley, California 95066-4922
United States of America.
Tel (408) 438 2900
Fax: (408) 438 5576/6231

Internet: <http://www.gpsemi.com>
CUSTOMER SERVICE CENTRES

- **FRANCE & BENELUX** Les Ulis Cedex Tel: (1) 69 18 90 00 Fax : (1) 64 46 06 07
- **GERMANY** Munich Tel: (089) 419508-20 Fax : (089) 419508-55
- **ITALY** Milan Tel: (02) 6607151 Fax: (02) 66040993
- **JAPAN** Tokyo Tel: (03) 5276-5501 Fax: (03) 5276-5510
- **KOREA** Seoul Tel: (2) 5668141 Fax: (2) 5697933
- **NORTH AMERICA** Scotts Valley, USA Tel: (408) 438 2900 Fax: (408) 438 5576/6231
- **SOUTH EAST ASIA** Singapore Tel:(65) 3827708 Fax: (65) 3828872
- **SWEDEN** Stockholm Tel: 46 8 702 97 70 Fax: 46 8 640 47 36
- **TAIWAN, ROC** Taipei Tel: 886 2 25461260 Fax: 886 2 27190260
- **UK, EIRE, DENMARK, FINLAND & NORWAY**
Swindon Tel: (01793) 726666 Fax : (01793) 518582

These are supported by Agents and Distributors in major countries world-wide.

© Mitel Corporation 1998 Publication No. AN59 Issue No. 2.0 July 1993

TECHNICAL DOCUMENTATION – NOT FOR RESALE. PRINTED IN UNITED KINGDOM

This publication is issued to provide information only which (unless agreed by the Company in writing) may not be used, applied or reproduced for any purpose nor form part of any order or contract nor to be regarded as a representation relating to the products or services concerned. No warranty or guarantee express or implied is made regarding the capability, performance or suitability of any product or service. The Company reserves the right to alter without prior notice the specification, design or price of any product or service. Information concerning possible methods of use is provided as a guide only and does not constitute any guarantee that such methods of use will be satisfactory in a specific piece of equipment. It is the user's responsibility to fully determine the performance and suitability of any equipment using such information and to ensure that any publication or data used is up to date and has not been superseded. These products are not suitable for use in any medical products whose failure to perform may result in significant injury or death to the user. All products and materials are sold and services provided subject to the Company's conditions of sale, which are available on request.

All brand names and product names used in this publication are trademarks, registered trademarks or trade names of their respective owners.