Index

Accumulated error metric, 490
Accumulation and buffering module, in LPC speech synthesis, 496
Acoustic direction tracker, 513–515
adaptc project, 332–334
adaptIDFIR project, 339–343
adaptIDFIRw project, 343
Adaptive channel equalization, adaptive structures for, 321–322
Adaptive filters, 154, 319–353. See also Adaptive FIR filter
    adaptive linear combiner, 319, 324–327
    adaptive structures in, 321–324
    performance function of, 320–321, 327–329
    search for minima of, 329–332
    for sinusoidal noise cancellation, 335
two-weight, 326–327
Adaptive FIR filter
    for noise cancellation, 335–339
    for system ID of fixed FIR, 339–343
    for system ID of fixed IIR, 343–345
Adaptive linear combiner, 319, 324–327
Adaptive predictor, adaptive structures for, 322–323
Adaptive temporal attenuator (ATA), 519–520
adaptnoise_2IN project, 335–339
adaptnoise project, 335
ADC, see Analog-to-digital converter
Add-compare-select operation, 490
Add instructions, 113
Additive white Gaussian noise (AWGN), for soft decision, 484
Addressing modes, linear and circular, 110–112
Address mode register (AMR), 111, 530
AES encryption standard, 503
AIC23 codec, 1, 20, 46–47
    changing LINE IN gain of, 50
    format of data, to and from, 52
    identification of bandwidth of, 85
    impulse response of, 89, 92
    settings defined in c6713dskinit.h, 33–34, 50
Aliasing, 74–75, 81–82, 82–85
    in impulse invariance method, 230
aliasing project, 82–85
Amplitude modulation (AM), 92–95,
    470–474
am project, 92–95
Analog-to-digital converter (ADC), 4, 34, 46, 102
Animation, 29
Antialiasing filter, 82
Arithmetic operations, of TMS320C6x processors, 105
asm statement, within C, 117
ASM (assembly) code
optimizing, 354–373
syntax of, 112–113
TMS320C6x format for, 112
ASM (assembly) functions. See also Linear assembly function
ASM program calling, 135–139
C calling, 131–135, 197–206, 251–252
Assembler directives, 115–116
Assembler optimizer, 355
Audio effects, 451–453
Automatic speaker recognition, 496–500
average project, 165–167
averagen project, 168
Bandpass FIR filters, 161, 175–177, 343
implementation of, 188
Bandstop FIR filters, 161, 175
implementation of, 188
Beat detection, using onboard LEDs, 429–434
beatdetector project, 434
Bilinear transformation (BLT), 217–220
design procedure using, 219
design using, 232–236
frequency warping in, 233
implementation method, 232
Binary phase-shift keying (BPSK), 454, 468–470, 474, 476
demodulation, 469–470
encoding/decoding, 468–469
modulation, 468–469
single-board transmitter/receiver simulation, 455–458
Binary representation, 533–539
bios_fastconv_TSK project, 386
bios_fft128c_SWI project, 385–386
bios_fir_SWI project, 383–385
bios_LED project, 376–378
bios_sine8_intr_LOG project, 382
bios_sine8_intr project, 379–381
Bit reversal, 268
Blackman window function, 164
BPSK folder, 468
BPSK_sim project, 455–468
BPSK transmitter/receiver, with PLL, 465–468
BPSK transmitter/voice encoder, with real-time input, 459–460
Branch instructions, 115
Branch metrics, 490
Build options, for CCS, 13–15
Butterfly graph, 260
C6416 folder, 7, 562
C6713 folder, 7
c6713dsk.cmd linker command file, 12, 34, 41
c6713dskinit.c initialization/communication file, 12, 31–33, 48
c6713dskinit.h header file, 33, 35–38
Cascade IIR filter structure, 215–216
C compiler, with CCS, 3, 6.
cdd flashburn utility, 99
Circular addressing, registers for, 530
Circular addressing mode, 111–112
Circular buffers, 111, 530
ASM functions with, 201–205
in external memory, 205–206
Classification module, in automatic speaker recognition, 497
Closed set identification, 498
Cluster, 497
Codebook, 497, 500
code_casm project, 135
Code Composer Studio (CCS), 1, 3, 6
build options for, 13–15
graphical displays with, 23
installation and support for, 6
memory window, 23, 26
Codec, see AIC23 codec
Code detection, using C calling an ASM function, 135
Code optimization, 354–373
compiler options for, 355–356
execution cycles for, 372–373
procedure for, 356
programming examples using, 356–363
software pipelining for, 363–372
steps in, 355–356
Codevectors, 500
Codeword, 497, 499
comm_intr() function, 33
comm_poll() function, 33, 48
Index

Compiler options
with CCS, 13–15
for code optimization, 355–356

Constellation diagram, 118, 500

Convolution, 146, 153, 165
frequency-domain, 306–312
time-domain, 299–306
Convolutional encoding, 482, 483

CPU functional units, of TMS320C6x processors, 105–106

Cross-path constraints, 128
Cross-path instructions, 127
Cross-paths, of TMS320C6x processors, 106

DAC, see Digital-to-analog converter

Data alignment, 123
Data allocation, 122–123
Data types, 124–125

Decimation, 508
Decimation-in-frequency (DIF) algorithms, 255
eight-point FFT using, 261–263
radix-2, 257–263
Decimation-in-time (DIT) algorithms, 255
eight-point FFT using, 267–268
radix-2, 263–268

Decode stage, 109
delay project, 53

Dependency graph, 363, 364–365
detect_play project, 453–454
dft128c project, 285–290
dft project, 273–277
dft.c program, 274, 564
dftw.c program, 277–278
fft.c program, 279, 564

Difference equations, 150–151
DTMF tone generation using, 247
sine generation using, 244–247
swept sinusoid generation using, 248–251

Digital-to-analog converter (DAC), 4, 46
dimpulse project, 78
DIP switch, 4, 9, 479–482
Direct form I IIR filter structure, 212
Direct form II IIR filter structure, 212–213, 223–225
Direct form II transpose IIR filter structure, 214–215
Direct memory access (DMA), 122
Discrete cosine transform (DCT), 255

Discrete Fourier transform (DFT), 255
of real-number sequence, 273–278
of real-time signal, 285–290

Discrete Hartley transform, 255
Division operation, 126
dotpia project, 135–139
dotp4clasm project, 139–141
dotp4 project, 23–30
dotpinitrinsic project, C code with, 358
dotpipedefloat project, ASM code with, 367

dotpipedefloat project, ASM code with, 367–372
dotpnpfloat project, ASM code with, 361–362
dotpnp project, ASM code with, 359–360
dotppfloat project, ASM code with, 362
dotpp project, ASM code with, 360

Dot product, 23–30
code optimization examples using, 357–372
double data type, 125
Double-precision (DP) data format, 125

DSK board, 3–5. See also DSP Starter Kit (DSK)
dsk_fir67.m program, 179
dsk_sos_iir67int.m program, 543
dsk_sos_iir67.m program, 240

DSP applications/student projects, 422–527
acoustic direction tracker, 513–515
adaptive temporal attenuator, 519–520
audio effects, 451–453
automatic speaker recognition, 496–500
beat detection using onboard LEDs, 429–434
binary phase shift keying, 468–470
convolutional encoding and Viterbi decoding, 482–492
dual-tone multifrequency signal detection, 422–429
encryption, 503–506
filter coefficient transfer, 434–435
filter design and implementation, 521
four-channel multiplexer, 522
FSK modem, 521
G.722 audio coding implementation, 501–502
IIR filter and scrambling scheme, 479–482
image processing, 521
modulation schemes, 470–479
mu-law for speech companding, 500–501
multirate filter, 508–513
DSP applications/student projects
(Continued)
normal network for signal recognition, 515–519
phase-locked loop project, 506–508
phase shift keying, 454–468
PID controller, 522
radix-4 FFT with RTDX using Visual C++ and MATLAB for plotting, 435–438
spectrum display, 438–445
speech synthesis, 493–496
time-frequency analysis of signals, 445–451
video line rate analysis, 522
voice detection and reverse playback, 453–454
DSP/BIOS, 374–421
hardware interrupts (HWI), 375, 378–381
idle functions (IDL), 375
periodic functions (PRD), 375, 376–378
software interrupts (SWI), 375, 382–386
tasks (TSK), 375, 386
threads, 374
DSP Starter Kit (DSK), 1. See also DSK entries
quick tests of, 7–9
DTMF_BIOS_RTDX project, 427–429
DTMF folder, 424
DTMF generation
using difference equations, 247
using lookup tables, 66–69
.D (data transfer) units, of TMS320C6x processors, 105
echo_control project, 54–57
echo project, 53–54
Edge detection, in image processing, 521
Eight-level PAM lookup table, 472
EMIF_LCD project, 440–445
Encoding regions, 500
Encryption, using data encryption standard algorithm, 503–506
encryption project, 503–506
Euclidean distances, speaker identification using, 500
Execute packets (EPs), 106
multiple, 129
Execute stage, 109
External memory, using to record voice, 95
External memory interface (EMIF), in spectrum display, 438–445
factclasm project, 141–142
factorial project, 132–133
f declaration, 124
fastconvdemo project, 306–308
Fast convolution, 297–317
with overlap-add for FIR implementation, 308–312
with overlap-add simulation for FIR, 306–308
fastconv project, 308–312
Fast Fourier transform (FFT), 255–318. See also Inverse fast Fourier transform (IFFT)
bit reversal and, 268
butterfly, 260
decimation-in-frequency algorithm for, 255
decimation-in-time algorithm with radix-2, 263–268
eight-point using decimation-in-frequency, 261–263
eight-point using decimation-in-time, 267–268
radix-2, 256–257
radix-4, 269–272
of real-time input signal, 290
using an FFT function in C, 290
using TI optimized radix-2 function, 295–297
using TI optimized radix-4 function, 297
of a sinusoidal signal, 290–295
sixteen-point, 263, 270–272
Fast Hartley transform (FHT), 255, 550–556
FDATool filter designer, 178, 540–544
for bandpass IIR filter design, 241
for FIR filter design, 541–542
for IIR filter design, 542–544
Feature extraction module, in automatic speaker recognition, 497
Fetch packets (FPs), 106
multiple EPs in, 129
fft128c project, 290
FFTr2 project, 295–297
FFTr4 project, 297
FFTsine table project, 290–295
f-function, in encryption, 504
File types, with CCS, 7
Finite impulse response (FIR) filters, 146–209. See also FIR entries; Infinite impulse response (IIR) filters
discrete signals and, 151–152
implementation using Fourier series, 158–162
with internally generated pseudorandom noise, 182–186
lattice structure of, 154–158
lowpass, 160, 161–162, 186–188
operation and design of, 152–154
programming examples using C and ASM code, 165–207
with RTDX using Visual C++ for filter coefficient transfer, 434–435
window functions for, 162–164
FIR2ways project, 193
FIR3LP project, 186–188
FIR4types project, 188–191
FIR bandpass filters, see Bandpass FIR filters
FIR bandstop filters, see Bandstop FIR filters
FIRcasmfast project, 200–201
FIRcasm project, 197–200
FIRcirc_ext project, 205–206
FIRcirc project, 201–205
FIR filter implementation
two different methods for, 193
using C calling a faster ASM function, 200–201
using C calling an ASM function, 197–200
with circular buffer, 201–205
with circular buffer in external memory, 205–206
using Fourier series, 158–162
using frequency-domain convolution, 306–308
using real-time frequency-domain convolution, 308–312
using real-time time-domain convolution, 299–306
using time-domain convolution, 306
FIRPRNbuf project, 182–186
FIRPRN project, 178–182
FIR project, 175–178
Fixed-point format, 124–125
Fixed-point operations, instructions for, 528–529
Fixed-point processors, 103
Flashburn (.cd) utility, 99
Flash memory, 105
erasing and programming, 99
using, 95–101
flash_sine project, 95–101
float data type, 125
Floating-point format, 125–126
Fourier series, FIR implementation using, 158–162
Four-level PAM lookup table, 472
Fractional fixed-point representation, 536
Frame-based processing, 280–297
Frame blocking, in determining MFCCs, 498
frames project, 283–284
Frame synchronization, 467
Frequency-domain plot, 25
Frequency inversion, scrambling by, 193
Frequency warping, 233
Functional unit latency, 110
Functional units, of TMS320C6x processors, 105–106
G.722 audio coding, implementation of, 501–502
G722 project, 501–502
Gabor expansion, 446
GEL slider, 18–19, 63–65
General Extension Language (GEL), 18
Goertzel algorithm, 422–429, 557–560
Goldwave shareware utility, 548–549
graphicEQ project, 312–317
Graphic equalizer, 312–317
graphic_FFT project, 438–440
Graph Property Dialogs, 24–25
Gray encoding, 475
Hamming distance, 490
Hamming window function, 163, 490, 495
Hanning window function, 163
Hardware interrupts (HWI), in DSP/BIOS, 375, 378–381
Header files, 7
hex6x.exe program, 98
.hex file, 98–99
Highpass FIR filter, 161
implementation, 188
Histogram equalization, in image processing, 521
HWI (DSP/BIOS Hardware interrupt), 375, 378–381
IDL (DSP/BIOS Idle function), 375
Idle functions (IDL), in DSP/BIOS, 375
**IIR_ctrl** folder, 482
IIR filter scheme, using onboard DIP switches, 479–482
iir project, 243–244
iirsoasadapt project, 343–345
iirsoedelta project, 228–230
iirsosprn project, 225–228
iirsos project, 223–225
iirsosstr.c program, 225
Image processing, 521
Impulse invariance method, 220–223
Impulse response, 78, 89, 92
Indirect addressing, 110
Infinite impulse response (IIR) filters, 210–254
second order sections, 216, 223
Initialization/communication file, 30–33
input_left_sample() function, 33, 48
Instruction sets, for TMS320C6x processors, 112–115
int data type, 124
Interrupt acknowledgment (IACK), 120
Interrupt clear register (ICR), 118, 530
Interrupt control registers, 118, 530
Interrupt enable register (IER), 118, 530
Interrupt flag register (IFR), 118, 530
Interrupt return pointer (IRP), 119
Interrupts, 118–121
registers for, 118, 530
Interrupt service table (IST), 120
Interrupt service table base (ISTB) register, 120
Interrupt service table pointer (ISTP), 118, 530
Interrupt set register (ISR), 118, 530
intrinsics, 126
INUMx signals, 120
Inverse discrete Fourier transform (IDFT), 255, 272
Inverse fast Fourier transform (IFFT), 272
Kaiser window function, 164
k-parameters, 154
LabVIEW, for PC/DSK interface, 415, 421
Laplace transform, 146, 147, 149, 221, 222
Lattice structure, of FIR filters, 154–158
Least mean squares (LMS) algorithms
for adaptive filters, 321, 323–324
sign-data algorithm, 323
sign-error algorithm, 323
sign-sign algorithm, 324
types of, 323–324
LED, 4, 9
Level detection, in determining MFCCs, 498
Levinson–Durbin algorithm, in LPC speech synthesis, 495
Linear adaptive combiner, see Adaptive linear combiner
Linear addressing mode, 110
Linear assembly, 116–117
Linear assembly function, C function calling, 139–142
Linear phase, with FIR filters, 154
Linear prediction, of speech signals, 493–496
Linear predictive coding (LPC), 493–494
Linker command files, 34–38
Linker options, with CCS, 15
Liquid-crystal displays (LCDs), in spectrum display, 440–445
Load instructions, 114–115
LOG object, in DSP/BIOS, 382
Lookup table
  DTMF generation with, 66–69
  impulse generation with, 78
  sine wave generation with, 60
  square-wave generation with, 75–77
Loop count, trip directive for, 127
loop_buf project, 57–60, 82
loop_intr project, 51–53, 564
loop_poll project, 48–51
Loop program
  using C6416 DSK, 564
  with input data stored in memory, 57–60
  using interrupts, 51–53, 564
  using polling, 48–51
Lowpass FIR filter, 160
  implementation, 188
.L (logical) units, of TMS320C6x processors, 105
MATLAB, 540–548
MATLAB student version
  for FIR filter design, 544–545
  for IIR filter design, 546–548
Mean-squared error, 328
Median filtering, in image processing, 521
Mel-frequency cepstrum coefficients (MFCCs), 497
Pseudorandom noise, 78–82
   as input to FIR filter, 178–186
   as input to IIR filter, 225–228
   as input to moving average filter, 168
   prandom.c program, 78–82
   using C calling an assembly function, 133–135
   PSK folder, 454–468
   PSOLA (pitch synchronous overlap-add) digital technology, 493
   Pulse amplitude modulation (PAM), 470–474
   Quadrature mirror filter (QMF), 501
   Quadrature phase-shift keying (QPSK), 474, 476
   Quantization error, 103
   Radix-2 decimation-in-frequency FFT algorithm, 255
   Radix-2 decimation-in-time FFT algorithm, 263–268
   Radix-2 fast Fourier transform, 256–257
   Radix-4 fast Fourier transform, 269–272
      of real-time input, 297
      with RTDX using Visual C++ and MATLAB for plotting, 406–411
      sixteen-point, 263, 270–272
   ramp project, 92
   Real-time data exchange (RTDX), 6
      for amplitude control of loop program output, 413–414
      for controlling generated sinusoid amplitude, 420–421
      for controlling generated sinusoid gain, 417–420
      displaying detected DTMF signals with, 427–429
      in filter coefficient transfer, 434–435
      for FIR filter implementation, 396–400, 415–417
      MATLAB–DSK interface using, 388–393
      with MATLAB FFT and plotting functions, 406–411
      for sine wave amplitude control, 400–406, 411–413
      spectrograms with, 446–450
      using LabVIEW for PC/DSK interface, 415–421
      using MATLAB for PC/DSK interface, 386–399
   using Visual Basic for PC/DSK interface, 411–414
   using Visual C++ to interface with DSK, 400–411
   Real-time scheduler, 374
   Real-time signal processing, 2
   Reconstruction filter, 46, 76–82, 145
   receiver folder, 465
   record project, 95
   Rectangular window function, 162
   Reflection coefficients, 154. See also $k$-parameters
   Register files, 110
   Registers
      for circular addressing and interrupts, 111, 530–532
      in indirect addressing, 110
      interrupt control, 118–120
      supporting data communication, 121
      for TMS320C6x processors, 110
   RELP (residue excited) digital technology, 493
   Remez exchange algorithm, 164
   Residual signal module, in LPC speech synthesis, 495
   Rijndael algorithm, 503
   rtdx_lv_filter project, 415–417
   rtdx_lv_gain project, 417–420
   rtdx_lv_sine project, 420–421
   rtdx_matlabFFT project, 393–396
   rtdx_matlabFIR project, 396–400
   rtdx_matlab_sim project, 388–393
   rtdx_vbloop project, 413–414
   rtdx_vbsine project, 411–413
   rtdx_vc_FFTmatlab folder, 406
   rtdx_vc_FFTTr4 project, 435–438
   rtdx_vc_FIR project, 434–435
   rtdx_vc_sine project, 400–406
   Sample update, viewing in memory, 199
   Sampling rate, 46
   $S$-boxes, in encryption, 505
   Scheduling tables, 363, 365–372
   scrambler project, 193–196
   Segmentation module, in LPC speech synthesis, 495
   Serial port control register (SPCR), 33, 530
   Serial ports, multichannel buffered, 121
   short data type, 124
   Short time Fourier transform (STFT), 445
   Sigma–delta technology, 46
Signal recognition, neural network for, 515–519

Sign-data LMS algorithm, 323

signed int data type, 124

Sign-error LMS algorithm, 323

Sign-sign LMS algorithm, 324

sin1500MATL project, 70–72

sine2sliders project, 63–65

sine8_buf project, 21–23

sine8_intr project, 60–61

sine8_LED project, 9–21, 562–564

sine8_phase_shift project, 462

sineDTMF_intr project, 66–69

sinegenDE project, 244–247

sinegenDTMF project, 247

sinegen_table project, 69–70

sine_intr project, 61–62

sine_led_ctrl project, 72–73

sine_stereo project, 62–63

sine wave generation

real-time, 60

sin1500.c program, 70–72

sine2sliders.c program, 63–65

sine8_intr.c program, 60

sine_intr.c program, 61

stereo output, 62–63

sweep8000.c program, 65–66

using lookup table, 60

using sin() function call, 61

using values generated in program, 69–70

with DIP switch control, 72–73

Single-precision (SP) data format, 125

Sinusoidal noise cancellation, adaptive filter

for, 335

Sixteen-level PAM lookup table, 473

Sliders

for amplitude and frequency of sine wave, 63–65

GEL files for 18–19

Software interrupts (SWI), in DSP/BIOS, 375, 382–384


soundboard folder, 451–453

Speaker identification, 500

Speaker recognition, automatic, 496–500

speaker_recognition folder, 496–500

Spectral leakage, 275, 290

spectrogram folder, 445–451

spectrogram_rtdx_mtl project, 445–450

Spectrograms

with RTDX using MATLAB, 446–450

with RTDX using Visual C++, 450–451

time-frequency analysis of signals with, 445–451

Spectrum display

through EMIF using LCDs, 440–445

through EMIF using 32 LEDs, 438–440

speech_syn project, 493–496

Speech synthesis, using linear prediction of speech signals, 493–496

squarewave project, 75–77

Stalling effects, 130

Stereo codec, 4, 40–42

Stereo output, sine generation with, 62–63

Store instructions, 114–115

Subtract instructions, 113–114

Sum of products, 357

sum project, 131–132

.S units, of TMS320C6x processors, 105

Support files, with CCS, 30

communication, 30–33

header, 33

initialization, 30–33

linker command, 34, 41

vector, 34, 39–40

sweep8000 project, 65–66

sweepDE project, 248–251

SWI (DSP/BIOS Software interrupt), 375, 382–386

sysid project, 85, 168, 345–352

sysid16 project, 85–92

system identification

adaptive structures for, 322

of codec antialiasing and reconstruction filters, 85–92

Tasks (TSK), in DSP/BIOS, 375, 386

Text-to-speech systems, 493

Threads, in DSP/BIOS, 374

timeconv project, 306

timeconvdemo project, 299–306

Time-domain plot, 24

Time-frequency analysis, of signals, 445–451

TLV320AIC23 onboard stereo codec, 2, 4, 46–47

TMS320C62xx fixed-point processors, 4

TMS320C6416 digital signal processor, 1, 5

TMS320C6416 DSK, 1, 561–566

TMS320C64x processors, 4, 561–566
TMS320C6713 digital signal processor, 1, 4, 104
board for, 4
TMS320C67xx floating-point processors, 4
TMS320C6x instruction set, 528–529
TMS320C6x processors, 1–3, 102–145
addressing modes and, 110–112
architecture of, 104–105
asm statement and, 117
assembler directives and, 115–116
C callable assembly functions with, 117–118
code improvement for, 126–128
constraints with, 128–130
CPU functional units of, 105–106
direct memory access and, 122
fetch and execute packets and, 106–108
fixed- and floating-point format and, 124–126
instruction set for, 112–115, 528–529
interrupts and, 118–121
linear assembly and, 116–117, 139–142
memory with, 122–124
multichannel buffered serial ports with, 121–122
pipelining and, 108–110
register files and, 110
timers and, 118
transmitter folder, 465
Transmitter/receiver algorithm
for PAM, 473–474
for PSK, 475
Trellis diagram, 485
TSK (DSP/BIOS Task), 375, 386
twosumfix project, ASM code with, 360–361
twosumfloat project, 362–363
twosumlasmfix project, linear ASM code with, 358–359
twosumlasmfloat project, linear ASM code with, 359
twosum project, C code for, 357–358
Universal synchronous bus (USB) cable, 3
Unsharp masking, in image processing, 521
Variable Watch, implementing, 26
Vector files, 34
vectors_intr.asm file, 39
vectors_poll.asm file, 12, 40
VELOCITI architecture, 5, 106, 561
VELP (voice excited) digital technology, 493
Very-long-instruction-word (VLIW)
arquitecture, 5, 106
Video line rate analysis, 522
Visual Basic (VB), for PC/DSK interface, 411–414
Visual C++, for PC/DSK interface, 400–411
Viterbi decoding algorithm, 482, 485
hard-decision decoding setup, 482
soft decision decoding setup, 482–483
viterbi project, 482–492
Voice detection, 453–454
Voice recording, using external memory, 95
Voice scrambling, using filtering and modulation, 193–196
VQ distortion, 497
VQ process, 499
Watch window, monitoring, 18–20
Wigner–Ville distribution, 446
Window functions, 162–164
Blackman, 164
Kaiser, 164
Hamming, 163
Hanning, 163
rectangular, 162
Windowing, in LPC speech synthesis, 495
z-transform ($ZT$), 146–150, 220–222