

i2c Address list (continue'd)

Receive 29 bites for Recalling the Scene Memory

```
case 0xF4: // Chord Edit mode
    transition_ae = (value2 << 8) + value3; // tra_e      = transition_ae;
    attack5      = (value4 << 8) + value5; // atk5       = attackB5;
    LFO5         = (value6 << 8) + value7; // lfom5      = LFOB5;
    pot00he      = (value8 << 8) + value9; // lpf_e      = pot00he;
    pot00ie      = (value10 << 8) + value11; // lpfSpd_e   = pot00ie;
    pot00je      = (value12 << 8) + value13; // lpf2_e     = pot00je;
    pot00ke      = (value14 << 8) + value15; // lpf2Spd_e  = pot00ke;
    pot00le      = (value16 << 8) + value17; // lpfWf_e    = pot00le;
    pot00me      = (value18 << 8) + value19; // lpf2Wf_e   = pot00me;
    pot00re      = value20;                // fdbk_e     = pot00ra;
    pot00se      = value21;                // cdpt_e     = pot00sa;
    pot00ne      = value22;                // lpfSW_e    = pot00ne;
    pot00oe      = value23;                // lpf2SW_e   = pot00oe;
    exB5         = value24;                // ex5        = exB5;
    op3SelectBe  = value25;                // op3_e      = op3SelectBe;
    arp2e        = value26;                // arptn_e    = arp2e;
    arpSpdBe     = value27;                // spd_e      = arpSpdBe;
    arpNotes_ae  = value28;                // apnote_e   = arpNotes_ae;
    addr1ae      = value29;                // waves_e    = addr1ae;
    break;

case 0xF5: // Sequencer mode
    transition_a1 = (value2 << 8) + value3; // tra_a1     = transition_af1;
    attack6       = (value4 << 8) + value5; // atk6       = attackB6;
    LFO6          = (value6 << 8) + value7; // lfom6      = LFOB6;
    pot00hf1      = (value8 << 8) + value9; // lpf_f1     = pot00hf1;
    pot00if1      = (value10 << 8) + value11; // lpfSpd_f1  = pot00if1;
    pot00jf1      = (value12 << 8) + value13; // lpf2_f1    = pot00jf1;
    pot00kf1      = (value14 << 8) + value15; // lpf2Spd_f1 = pot00kf1;
    pot00lf1      = (value16 << 8) + value17; // lpfWf_f1   = pot00lf1;
    pot00mf1      = (value18 << 8) + value19; // lpf2Wf_f1  = pot00mf1;
    pot00rf1      = value20;                // fdbk_f1    = pot00ra;
    pot00sf1      = value21;                // cdpt_f1    = pot00sa;
    pot00nf1      = value22;                // lpfSW_f1   = pot00nf1;
    pot00of1      = value23;                // lpf2SW_f1  = pot00of1;
    exB6          = value24;                // ex6        = exB6;
    op3SelectBf1  = value25;                // op3_f1     = op3SelectBf1;
    arp2f1        = value26;                // arptn_f1   = arp2f1;
    arpSpdBf1     = value27;                // spd_f1     = arpSpdBf1;
    arpNotes_af1  = value28;                // apnote_f1  = arpNotes_af1;
    addr1af1      = value29;                // waves_f1   = addr1af1;
    break;
```