INTRODUCTION
This application note describes the differences between the ADE7878 and the ADE7880. It discusses the hardware and software differences and provides a header file for the ADE7880.

HARDWARE DIFFERENCES
The ADE7880 is pin-for-pin compatible with the ADE7878. New Antialiasing Filters
However, because the ADC bandwidth has been increased from 2 kHz (−3 dB point) to 3.3 kHz (−3 dB point), the antialiasing filters used in the input datapath of the ADCs has to be changed. Previously, on the ADE7878 evaluation board, a 1 kΩ/22 nF (7.2 kHz corner) antialiasing filter was used. A 1 kΩ/10 nF (15.9 kHz corner) antialiasing filter is used on the ADE7880 evaluation board.

NEUTRAL CURRENT MAY USE DIFFERENT SENSOR THAN PHASE CURRENTS
The neutral current may be sensed using a different sensor type than the phase currents. For example, the phase currents may be sensed with Rogowski coils and the neutral current may be sensed with current transformers (CTs), or vice versa.
Use Bit 0 (INTEN) in the CONFIG register of the ADE7880 to enable/disable the integrators in the phase current channels.
Use Bit 3 (ININTEN) in the CONFIG3 register of the ADE7880 to enable/disable the integrator in the neutral current channel.
The definition of the gain register at Address 0xE60F in the ADE7880 has remained the same.
TABLE OF CONTENTS
Hardware Differences .......................................................... 1
Neutral Current May Use Different Sensor than Phase Currents.... 1
Revision History ........................................................................ 2
Software Differences ............................................................. 3
Appendix: ADE7880.H Header File............................................5

REVISION HISTORY
3/12—Rev. 0 to Rev. A
Changes to Table 2..................................................................... 4

11/11—Revision 0: Initial Version
SOFTWARE DIFFERENCES

Register Addresses

The register map has changed. Many of the ADE7878 registers now have new addresses. The ADE7880 has additional registers because of the new harmonic calculations. The register information is found in the Appendix: ADE7880.H Header File section.

ADE7880 Does Not Compute the Total Reactive Power and the Total Reactive Energy

The ADE7878 computes the total and fundamental reactive powers/energies. The ADE7880 computes only the fundamental reactive power/energy.

The ADE7878 stores the instantaneous values of the phase total reactive powers into AV AR, BV AR, and CV AR registers. These registers have been eliminated.

The ADE7878 HSDC port transmits the phase total reactive powers when Bits [4:3] (HXFER) in the HSDC_CFG register have been set to 10. Instead, the ADE7880 transmits the fundamental reactive powers when the HXFER bits have been set to 10. The ADE7880 phase fundamental reactive power registers, AFV AR, BFV AR and CFV AR, are not mapped with an address in the register space and can be accessed only through the HSDC port.

xPGAIN Registers Replaced the xWGAIN, xVARGAIN, xVAGAIN Registers (x = A, B, C)

In the ADE7878, the gain registers in the active, reactive, and apparent powers datapaths were AWGAIN, BWGAIN, CWGAIN, AVARGAIN, BVARGAIN, CVARGAIN, AVAGAIN, BVAGAIN, and CVAGAIN. The recommendation was to use the same values to initialize them on each phase.

In the ADE7880, the APGAIN, BPAGAIN, and CPAGAIN registers replace the xWGAIN, xVARGAIN, and xVAGAIN on each phase. APGAIN manages all the power gains on Phase A, BPAGAIN manages all the power gains on Phase B, and CPAGAIN manages all the power gains on Phase C.

The WTHR, VARTH, and VATHR Register Definitions Changed

In the ADE7878, the WTHR, VARTH, and VATHR, 48-bit registers are defined as

\[
WTHR = VARTH = VATHR = \frac{P_{MAX} \times f_s \times 3600 \times 10^n}{U_{FS} \times I_{FS}}
\]

where:

\(P_{MAX} = 33,516,139 = 0x1FF6A6B\) as the instantaneous power computed when the ADC inputs are at full scale.

\(f_s = 8\ \text{kHz},\) the frequency with which the DSP computes the instantaneous power.

\(n\) is an integer that determines what derivative of \(\text{wh}[10^n\ \text{wh}]\) is desired as one LSB of the xWATTHR/xVARHR/xVAHR registers.

\(U_{FS}\) and \(I_{FS}\) are the rms values of phase voltages and currents when the ADC inputs are at full scale.

In the ADE7880, the WTHR, VARTH, and VATHR are now 8-bit unsigned registers and are defined as

\[
WTHR = VARTH = VATHR = \frac{P_{MAX} \times f_s \times 3600 \times 10^n}{U_{FS} \times I_{FS} \times 2^{27}}
\]

where:

\(P_{MAX} = 27,059,678 = 0x19CE5DE\) as the instantaneous power computed when the ADC inputs are at full scale.

\(f_s = 1.024\ \text{MHz},\) the frequency at which every instantaneous power computed by the DSP at 8 kHz is accumulated.

\(n\) is an integer that determines what derivative of \(\text{wh}[10^n\ \text{wh}]\) is desired as one LSB of the xWATTHR/xVARHR/xVAHR registers.

\(U_{FS}\) and \(I_{FS}\) are the rms values of phase voltages and currents when the ADC inputs are at full scale.

No Load Management Changed for the Total Active/ Reactive and the Fundamental Active/Reactive Powers

In the ADE7880, the no load condition for the total active/reactive powers and the fundamental active/reactive powers has changed. See the ADE7880 data sheet for more information.

HPF Managed by Bit 0 (HPFEN) in the CONFIG3 Register (ADE7880)

In the ADE7878, the high-pass filters (HPFs) used in the current and voltage channels datapaths are managed by the HPFDIS 24-bit register. If the register is 0, its default value, the HPFs are enabled. If the register is initialized with a nonzero value, the HPFs are disabled.

In the ADE7880, the HPFs are managed by Bit 0 (HPFEN) in the CONFIG3 register. If HPFEN is 0, the HPFs are enabled. If HPFEN is 1, its default value, the HPFs are enabled.

ADE7880 Computes RMS Value of the Sum of the Phase Currents

The ADE7878 and the ADE7880 compute the instantaneous value of the sum of the phase currents and store it into the ISUM register. The ADE7880 does not compute the rms of ISUM.

The ADE7880 computes the rms of ISUM and stores it into the NIRMS register if Bit 2 (INSEL) of the CONFIG3 register (see the ADE7880 data sheet) is set to 1. If INSEL is 0, its default value, the NIRMS register contains the rms of the neutral current sensed at the INP and INN pins.

ADE7880 Computes RMS of Third Voltage in 3P3W Configurations

In 3P3W configurations (when the CONSEL bits in the ACCMODE register are set to 01), only Phase A and Phase C are sensed using Phase B as reference. Both the ADE7878 and the ADE7880 compute the rms values of the line voltages between Phase A and Phase B and between Phase C and Phase B and store them into the AVRMS and CVRMS registers. The ADE7880 computes the rms values of the line voltage between Phase A and Phase C and stores them into the BVRMS register.
ADE7880 May Compute Smoother Instantaneous Active Powers

Bit 1 (LPFSEL) in CONFIG3 register manages the settling time of the total active power calculations in the ADE7880. If LPFSEL is cleared to 0, its default value, the settling time is 650 ms. If LPFSEL is set to 1, the settling time is 1300 ms, providing for smoother instantaneous total active power.

ADE7880 Introduces Communication Verification Registers

The ADE7880 includes a set of three registers that allow any communication via I2C or SPI to be verified. The LAST_OP, LAST_ADD, and LAST_RWDATA registers record the nature, address, and data of the last successful communication, respectively.

ADE7880 Improves CHECKSUM Calculations

In the ADE7878, the CHECKSUM calculations cover 13 configuration registers and 6 internal registers.

In the ADE7880, the CHECKSUM calculations cover 13 configuration registers, all registers located in the DSP data memory RAM between Address 0x4380 and Address 0x43BE and 8 internal registers. In addition, every time a register is written or changes value inadvertently, Bit 25 (CRC) in the STATUS1 register is set to 1. If Bit 25 (CRC) in the MASK1 register is set to 1, the IRQ1 interrupt pin is driven low.

Conversion Modes Changed

In the ADE7878, the energy-to-frequency converter generates pulses at Pin CF1, Pin CF2, and Pin CF3/HSCLK function of Bits[1:0] (WATTACC) and Bits[3:2] (VARACC) in the ACCMODE register. The instantaneous powers are always signed accumulated in the energy registers independent of the state of the WATTACC and VARACC bits.

In the ADE7880, the modes determined by the WATTACC and VARACC bits in the ACCMODE register have changed (see Table 1 and Table 2). The instantaneous powers are now accumulated into the energy registers based on their state. The energy-to-frequency converter generates pulses at the CF1, CF2/HREADY, and CF3/HSCLK pins also based on the WATTACC and VARACC bits, with one exception. When the instantaneous total and fundamental active powers are accumulated in positive only mode (WATTACC = 01), the energy-to-frequency converter still generates the pulses in signed accumulation mode.

Identifying the ADE7880 and ADE7878 when the Same Board Can Accommodate Both of Them

The CONFIG register for both the ADE7880 and the ADE7878 is located at the same address, Address 0xE618. The default value is 0x0 for the ADE7878 and 0x2 for the ADE7880.

To identify if the ADE7878 or the ADE7880 is mounted on the board read Address 0xE618 after power up. If the value is 0x0, the ADE7878 is mounted. If the value is 0x2, the ADE7880 is mounted.

Energy Accumulation Modes and Energy-to-Frequency

Table 1. Total Active Power Accumulation Mode and Fundamental Active Power Accumulation Mode

<table>
<thead>
<tr>
<th>WATTACC Bits (Bits[1:0] in the ACCMODE Register)</th>
<th>Total/Fundamental Active Energy Registers Accumulation Modes</th>
<th>CF Pulse Generation Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Signed accumulation.</td>
<td>Same as energy registers</td>
</tr>
<tr>
<td>01</td>
<td>Positive only accumulation.</td>
<td>Signed accumulation</td>
</tr>
<tr>
<td>10 (reserved)</td>
<td>The ADE7880 behaves like WATTACC Bits[1:0] = 00.</td>
<td>Same as energy registers</td>
</tr>
<tr>
<td>11</td>
<td>Absolute accumulation.</td>
<td>Same as energy registers</td>
</tr>
</tbody>
</table>

Table 2. Fundamental Reactive Power Accumulation Modes

<table>
<thead>
<tr>
<th>VARACC Bits (Bits[3:2] in ACCMODE Register)</th>
<th>Fundamental Reactive Energy Registers Accumulation Modes</th>
<th>CF Pulse Generation Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Signed accumulation.</td>
<td>Same as energy registers</td>
</tr>
<tr>
<td>01 (Reserved)</td>
<td>The ADE7880 behaves like VARACC Bits[1:0] = 00.</td>
<td>Same as energy registers</td>
</tr>
<tr>
<td>10</td>
<td>The fundamental reactive power is accumulated depending on the sign of the fundamental active power. If the active power is positive, the reactive power is accumulated as is, whereas if the active power is negative, the reactive power is accumulated with reversed sign.</td>
<td>Same as energy registers</td>
</tr>
<tr>
<td>11</td>
<td>Absolute accumulation.</td>
<td>Signed accumulation</td>
</tr>
</tbody>
</table>
APPENDIX: ADE7880.H HEADER FILE

```c
#ifndef __ADE7880_H__
#define __ADE7880_H__

#define AIGAIN 0x4380
#define AVGAIN 0x4381
#define BIGAIN 0x4382
#define BVGAIN 0x4383
#define CIGAIN 0x4384
#define CVGAIN 0x4385
#define NIGAIN 0x4386
#define DICOEFF 0x4388
#define APGAIN 0x4389
#define AWATTOS 0x438A
#define BPGAIN 0x438B
#define BWATTOS 0x438C
#define CPgain 0x438D
#define CWATTOS 0x438E
#define AIRMSOS 0x438F
#define AVRMSOS 0x4390
#define NIRMOS 0x4391
#define CVRMSOS 0x4392
#define CIRMOS 0x4393
#define CVRMSOS 0x4394
#define NIRMOS 0x4395
#define HPgain 0x4396
#define HIZRMSOS 0x4397
#define BVgain 0x4398
#define BFWATTOS 0x4399
#define CFWATTOS 0x439A
#define AFVAROS 0x439B
#define BFVAROS 0x439C
#define CFVAROS 0x439D
#define AFVAROS 0x439E
#define BFVAROS 0x439F
#define CFVAROS 0x43A0
#define AFVAROS 0x43A1
#define BFVAROS 0x43A2
#define CFVAROS 0x43A3
#define AFVAROS 0x43A4
#define BFVAROS 0x43A5
#define CFVAROS 0x43A6
#define AFVAROS 0x43A7
#define BFVAROS 0x43A8
#define CFVAROS 0x43A9
#define AFVAROS 0x43AA
#define BFVAROS 0x43AB
#define CFVAROS 0x43AC
#define AFVAROS 0x43AD
#define BFVAROS 0x43AE
#define CFVAROS 0x43AF
#define AFVAROS 0x43B0
#define BFVAROS 0x43B1
#define CFVAROS 0x43B2
#define AFVAROS 0x43B3
```

`#define` followed by a variable name and a value is used to define constants in the header file. These constants are used to represent specific addresses or values in the ADE7880 chip.
#define VAWV 0xE510
#define VBWV 0xE511
#define VCWV 0xE512
#define AWATT 0xE513
#define BWATT 0xE514
#define CWATT 0xE515
#define AFVAR 0xE516
#define BFVAR 0xE517
#define CFVAR 0xE518
#define AVA 0xE519
#define BVA 0xE51A
#define CVA 0xE51B
#define CHECKSUM 0xE51F
#define VNOM 0xE520
#define LAST_RWDATA_24bit 0xE5FF
#define PHSTATUS 0xE600
#define ANGLE0 0xE601
#define ANGLE1 0xE602
#define ANGLE2 0xE603
#define PHNOLOAD 0xE608
#define LINECYC 0xE60C
#define ZXTOUT 0xE60D
#define COMPMODE 0xE60E
#define Gain 0xE60F
#define CFMODE 0xE610
#define CF1DEN 0xE611
#define CF2DEN 0xE612
#define CF3DEN 0xE613
#define APHCAL 0xE614
#define BPHCAL 0xE615
#define CPHCAL 0xE616
#define PHSIGN 0xE617
#define CONFIG 0xE618
#define MMODE 0xE700
#define ACCMODE 0xE701
#define LCYCMODE 0xE702
#define PEAKCYC 0xE703
#define SAGCYC 0xE704
#define CFCYC 0xE705
#define HSDC_CFG 0xE706
#define Version 0xE707
#define LAST_RWDATA_8bit 0xE7FD
#define FVRMS 0xE880
#define FIRMS 0xE881
#define FWATT 0xE882
#define FVAR 0xE883
#define FVA 0xE884
#define FPF 0xE885
#define VTHDN 0xE886
#define ITHDN 0xE887
#define HXVRMS 0xE888
#define HXWATT 0xE88A
#define HXVAR 0xE88B
#define HXVA 0xE88C
#define HXPF 0xE88D
#define HXVHD 0xE88E
#define HXIF 0xE88F
#define HYVRMS 0xE890
#define HYVAR 0xE893
#define HYVA 0xE894
#define HYPF 0xE895
#define HYVHD 0xE896
#define HYIHD 0xE897
#define HZVRMS 0xE898
#define HZWATT 0xE89A
#define HZVAR 0xE89B
#define HZVA 0xE89C
#define HZPF 0xE89D
#define HZVHD 0xE89E
#define HZIHG 0xE89F
#define HCONFUG 0xE900
#define APF 0xE902
#define BPF 0xE903
#define CPF 0xE904
#define APERIOD 0xE905
#define BPERRIOD 0xE906
#define CPERIOD 0xE907
#define APNOLOAD 0xE908
#define VARNLOAD 0xE909
#define VANLOAD 0xE90A
#define LAST_ADD 0xE9FE
#define LAST_RWDATA_16bit 0xE9FF
#define CONFIG3 0xEA00
#define LAST_OP 0xEA01
#define WTHR 0xEA02
#define VARTHR 0xEA03
#define VATHR 0xEA04
#define   HX_reg           0xEA08
#define   HY_reg           0xEA09
#define   HZ_reg           0xEA0A
#define   LPOILVL           0xEC00
#define   CONFIG2           0xEC01

#define   HX_reg           0xEA08
#define   HY_reg           0xEA09
#define   HZ_reg           0xEA0A
#define   LPOILVL           0xEC00
#define   CONFIG2           0xEC01

#endif /* __ADE7880_H__ */
I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).