ARM
CLIMATE RESEARCH FACILITY

Ka-Band ARM Zenith Radar (KAZR)
HANDBOOK

February 2012

U.S. DEPARTMENT OF ENERGY | Office of Science

DISCLAIMER

Ka-Band ARM Zenith Radar (KAZR)

K Widener
N Bharadwaj
K Johnson

February 2012

Acronyms and Abbreviations

ARM	Atmospheric Radiation Measurement (Climate Research Facility)
ARSCL	Active Remote Sensing of Clouds
C band	frequencies between 4 GHz and 8 GHz
dB	decibel
dBi	antenna gain referenced to isotropic radiator
dBm	decibel referenced to 1 mW
dBZ	reflectivity
DMF	Data Management Facility
DOE	U.S. Department of Energy
DQO	Data Quality Office (ARM)
GHz	gigahertz (10^9 Hz)
Hz	hertz
Ka band	frequencies between 26.5 GHz and 40 GHz
KAZR	Ka-band ARM Zenith Radar
K_{DP}	specific differential phase
kW	kilowatt
LNA	low noise amplifier
m	meter
MHz	megahertz (10^6 Hz)
MMCR	millimeter-wavelength cloud radar
MMW	millimeter wave (30GHz–300GHz)
mW	milliwatt
NSA	North Slope of Alaska
PNNL	Pacific Northwest National Laboratory (Battelle)
PRF	pulse repetition frequency
RF	radio frequency
SGP	Southern Great Plains
TWP	Tropical Western Pacific
TWT	Traveling Wave Tube
TWTA	Traveling Wave Tube Amplifier
VAP	value-added product

Contents

Figures

Tables

1.0   General Overview

1.1   Introduction

Figure 1. KAZR installation at SGP (foreground) with SACR (background).

The Ka-band ARM zenith radar (KAZR) is a zenith-pointing Doppler cloud radar operating at approximately 35 GHz. The KAZR is an evolutionary follow-on radar to ARM's widely successful millimeter-wavelength cloud radar (MMCR). The main purpose of the KAZR is to provide vertical profiles of clouds by measuring the first three Doppler moments: reflectivity, radial Doppler velocity, and spectra width. At the sites where the dual-polarization measurements are made, the Doppler moments for the cross-polarization channel are also available. In addition to the moments, velocity spectra are also continuously recorded for each range gate.

1.1.1   Transmitter

Type: TWTA

Center frequency: 35 GHz

Peak power output: ~ 200 watts

Pulse width: 100 ns–20 μs

Polarization: dual-polarization at the Southern Great Plains (SGP) and Barrow, single-polarization at Manus, Darwin, and the second ARM Mobile Facility (AMF2)

Maximum duty cycle: 25%

PRF: maximum 20 kHz

Manufacturer: Applied Systems Engineering

1.1.2   Receiver

Type: dual-channel digital

Dynamic range: > 80 dB

Noise figure:	4.5 dB
Sampling rate:	120 MHz
Decimation factor:	Adjustable
Video bandwidth:	Adjustable
Processing software:	custom development
Manufacturer:	Mercury Computers

1.1.3 Antenna

Antenna Type:	Cassegrain parabolic reflector
Diameter:	3 m at SGP, 2m at other sites
3 dB beam width:	0.2° at SGP, 0.3° at other sites
Gain:	57.5 dBi at SGP, 53 dBi at other sites
Cross polarization isolation:	- 27 dB
2-way radome loss:	< 2.0 dB
Manufacturer:	Millitech

2.0 Contacts

2.1 Mentor

Kevin Widener

Pacific Northwest National Laboratory

412 Hale Drive, Underwood, WA 98651

Phone: 509-528-9565

Email: kevin.widener@pnnl.gov

Nitin Bharadwaj

Pacific Northwest National Laboratory

P.O. Box 999 Richland, WA 99352

Phone: 509-372-4267 Fax: 509-375-6736

Email: nitin@pnnl.gov

Karen Johnson (for data issues)

Environmental Sciences Department

Brookhaven National Laboratory

Upton, NY 11973

Phone: 631-344-5952 Fax: 631-344-2060

Email: kjohnson@bnl.gov

2.2   Vendor/Instrument Developer

ProSensing, Inc.

107 Sunderland Road

Amherst, MA 01002

Website: http://www.prosensing.com

3.0   Deployment Locations and History

Table 1. KAZR locations.

Site	Location	Latitude	Longitude	Altitude (m)	Dual-Polarization?	Date Installed
SGP/C1	Billings, OK	36.605300 N	97.486294 W	316	Yes	01/18/2011
NSA/C1	Barrow, AK	71.322953 N	156.615839 W	7	Yes	05/12/2011
TWP/C1	Manus, PNG	2.061062 S	147.425297 E	4	No	03/12/2011
TWP/C3	Darwin, AUS	12.424518 S	130.891527 E	29.9	No	01/27/2011
AMF2	Gan, Maldives	0.690597 S	73.149897 E	3	No	09/02/2011

Figure 2. KAZR installation at Barrow.

3

Figure 3. KAZR installation on Manus Island, Papua New Guinea.

Figure 4. KAZR installation at Darwin, Australia.

4.0   Near-Real-Time Data Plots

See http://plot.dmf.arm.gov/plotbrowser/.

5.0 Data Description and Examples

KAZR moments data are available from the [ARM Data Archive](#) in the following datastreams: kazrbl, kazrci, and kazrge. The 'kazrge' files contain the less-sensitive but full-range burst pulse moments. The 'kazrbl' and 'kazrci' datastreams contain chirp pulse moments (see Section 5.2, Annotated Examples for more information on burst and chirp pulses). The 'kazrci' moments were collected with the longest chirp pulse, so are the most sensitive, but have the deepest 'blanked-out' region near the ground. The 'kazrbl' moments were collected with a shorter chirp pulse, so are slightly less sensitive than the kazrci moments, but their available range extends closer to ground level. Table 2 indicates which datastreams are available at each site and time.

Table 2. KAZR datastream availability.

Site	Kazrge	Kazrci	Kazrbl
NSA	20110512–present	20110512–20111109	20111112–present
SGP	20110118–present	20110118–20110503	20110503–present
TWP-C1	20110312–present	20110312–present	
TWP-C3	20110127–20110613 20111006–present	20110127–20110613 20111006–present	
AMF2	20111006–present	20111009–present	20111006–20111007

5.1 Data File Contents

The kazrge, kazrci, and kazrbl data files have the netCDF data header below. Note that only sites that make dual-polarization measurements, SGP and Barrow, will have both copolar ('copol') and cross-polar ('xpol') moments in their data files.

```
dimensions:
    time = UNLIMITED ; // (23386 currently)
    range = 561 ;
variables:
    int base_time ;
    base_time:string = "30-Nov-2011,23:00:05 GMT" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2011-11-30 23:00:05 0:00" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2011-12-01 00:00:00 0:00" ;
    int qc_time(time) ;
        qc_time:long_name = "Quality check results on field: Time offset from midnight" ;
        qc_time:units = "unitless" ;
```

qc_time:description = "This field contains bit packed values which should be interpreted as listed. No bits set (zero) represents good data." ;
qc_time:bit_1_description = "Delta time between current and previous samples is zero." ;
qc_time:bit_1_assessment = "Indeterminate" ;
qc_time:bit_2_description = "Delta time between current and previous samples is less than the delta_t_lower_limit field attribute." ;
qc_time:bit_2_assessment = "Indeterminate" ;
qc_time:bit_3_description = "Delta time between current and previous samples is greater than the delta_t_upper_limit field attribute." ;
qc_time:bit_3_assessment = "Indeterminate" ;
qc_time:delta_t_lower_limit = 1. ;
qc_time:delta_t_upper_limit = 10. ;
qc_time:prior_sample_flag = 1 ;
qc_time:comment = "If the \'prior_sample_flag\' is set the first sample time from a new raw file will be compared against the time just previous to it in the stored data. If it is not set the qc_time value for the first sample will be set to 0." ;
float range(range) ;
    range:long_name = "Range (center of radar sample volume)" ;
    range:units = "m" ;
    range:missing_value = -9999.f ;
float reflectivity_copol(time, range) ;
    reflectivity_copol:long_name = "Reflectivity, copolar" ;
    reflectivity_copol:units = "dBZ" ;
    reflectivity_copol:missing_value = -9999.f ;
float reflectivity_xpol(time, range) ;
    reflectivity_xpol:long_name = "Reflectivity, cross-polar" ;
    reflectivity_xpol:units = "dBZ" ;
    reflectivity_xpol:missing_value = -9999.f ;
float mean_doppler_velocity_copol(time, range) ;
    mean_doppler_velocity_copol:long_name = "Mean doppler velocity, copolar" ;
    mean_doppler_velocity_copol:units = "m/s" ;
    mean_doppler_velocity_copol:resolution = 0.001f ;
    mean_doppler_velocity_copol:missing_value = -9999.f ;
    mean_doppler_velocity_copol:positive_velocities = "Positive values indicate motion away from the radar." ;
    mean_doppler_velocity_copol:negative_velocities = "Negative values indicate motion toward from the radar." ;
float mean_doppler_velocity_xpol(time, range) ;
    mean_doppler_velocity_xpol:long_name = "Mean doppler velocity, cross-polar" ;
    mean_doppler_velocity_xpol:units = "m/s" ;
    mean_doppler_velocity_xpol:resolution = 0.001f ;
    mean_doppler_velocity_xpol:missing_value = -9999.f ;
    mean_doppler_velocity_xpol:positive_velocities = "Positive values indicate motion away from the radar." ;
    mean_doppler_velocity_xpol:negative_velocities = "Negative values indicate motion toward from the radar." ;
float spectral_width_copol(time, range) ;
    spectral_width_copol:long_name = "Spectral width, copolar" ;
    spectral_width_copol:units = "m/s" ;
    spectral_width_copol:resolution = 0.001f ;
    spectral_width_copol:missing_value = -9999.f ;

```
float spectral_width_xpol(time, range) ;
    spectral_width_xpol:long_name = "Spectral width, cross-polar" ;
    spectral_width_xpol:units = "m/s" ;
    spectral_width_xpol:resolution = 0.001f ;
    spectral_width_xpol:missing_value = -9999.f ;
float signal_to_noise_ratio_copol(time, range) ;
    signal_to_noise_ratio_copol:long_name = "Signal to noise ratio, copolar" ;
    signal_to_noise_ratio_copol:units = "dB" ;
    signal_to_noise_ratio_copol:resolution = 0.001f ;
    signal_to_noise_ratio_copol:missing_value = -9999.f ;
float signal_to_noise_ratio_xpol(time, range) ;
    signal_to_noise_ratio_xpol:long_name = "Signal to noise ratio, cross-polar" ;
    signal_to_noise_ratio_xpol:units = "dB" ;
    signal_to_noise_ratio_xpol:resolution = 0.001f ;
    signal_to_noise_ratio_xpol:missing_value = -9999.f ;
float cal_constant_copol(time) ;
    cal_constant_copol:long_name = "Radar calibration constant, copolar" ;
    cal_constant_copol:units = "dB" ;
    cal_constant_copol:missing_value = -9999.f ;
    cal_constant_copol:comment = "Values are updated on an approximately hourly basis, as new
    raw data files are collected." ;
float cal_constant_xpol(time) ;
    cal_constant_xpol:long_name = "Radar calibration constant, cross-polar" ;
    cal_constant_xpol:units = "dB" ;
    cal_constant_xpol:missing_value = -9999.f ;
    cal_constant_xpol:comment = "Values are updated on an approximately hourly basis, as new raw
    data files are collected." ;
float rx_noise(time) ;
    rx_noise:long_name = "Receiver noise" ;
    rx_noise:units = "dBm" ;
    rx_noise:missing_value = -9999.f ;
    rx_noise:comment = "Values are updated on an approximately hourly basis, as new raw data files
    are collected." ;
float roll(time) ;
    roll:long_name = "Side to side sway" ;
    roll:units = "degrees" ;
    roll:missing_value = -9999.f ;
float pitch(time) ;
    pitch:long_name = "Rise and fall (front to back)" ;
    pitch:units = "degrees" ;
    pitch:missing_value = -9999.f ;
float lat ;
    lat:long_name = "North latitude" ;
    lat:units = "degree_N" ;
    lat:valid_min = -90.f ;
    lat:valid_max = 90.f ;
float lon ;
    lon:long_name = "East longitude" ;
    lon:units = "degree_E" ;
    lon:valid_min = -180.f ;
    lon:valid_max = 180.f ;
```

```
float alt ;
    alt:long_name = "Altitude above mean sea level" ;
    alt:units = "m" ;
```

5.1.1   Primary Variables and Expected Uncertainty

Table 3. Primary variables.

Variable	Description	Uncertainty
reflectivity_copol	Co-polarization reflectivity (time, height) in dBZ	3 dBZ
reflectivity_xpol	Cross-polarization reflectivity (time, height) in dBZ	3 dBZ
mean_doppler_velocity_copol	Co-polarization radial Doppler velocity (time, height) in m/s	0.1 m/s
mean_doppler_velocity_xpol	Cross-polarization radial Doppler velocity (time, height) in m/s	0.1 m/s
spectral_width_copol	Co-polarization spectral width (time, height) in m/s	0.1 m/s
spectral_width_xpol	Cross-polarization spectral width (time, height) in m/s	0.1 m/s

5.2 Annotated Examples

The KAZR transmits and receives two different types of pulses: burst and chirp. The burst pulse is a simple pulse of RF energy. The chirp pulse is a frequency-modulated pulse that is longer than the burst pulse. This longer pulse provides for more energy to be transmitted and gives greater sensitivity than the burst pulse. Since the chirp pulse is modulated, we can still have range resolution similar to a simple burst pulse. There is a penalty for getting this extra sensitivity, and these are called range sidelobes.

Figure 5 shows a time-height plot of reflectivity. Note that the data go all the way down to ground level. However, in Figure 6, there is a "blanked" portion at the bottom of the plot. This is inserted to remove corruption of range sidelobes generated by the chirped pulse. Also note that there are more clouds visible in Figure 6 than in Figure 5, showing increased sensitivity when using the modulated pulse. For instance, compare the clouds from 0200–0300 at 8km to 10km.

8

KAZR Reflectivity (Copol) Burst Pulse
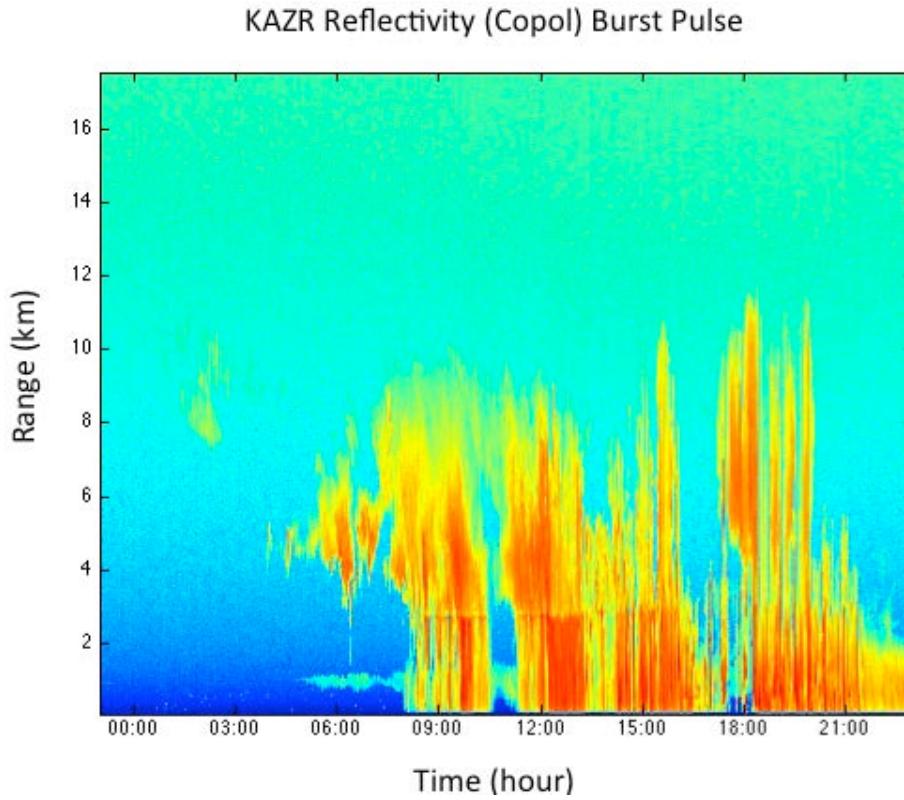
Figure 5. Burst pulse reflectivity plot.

KAZR Reflectivity (Copol) Chirped Pulse
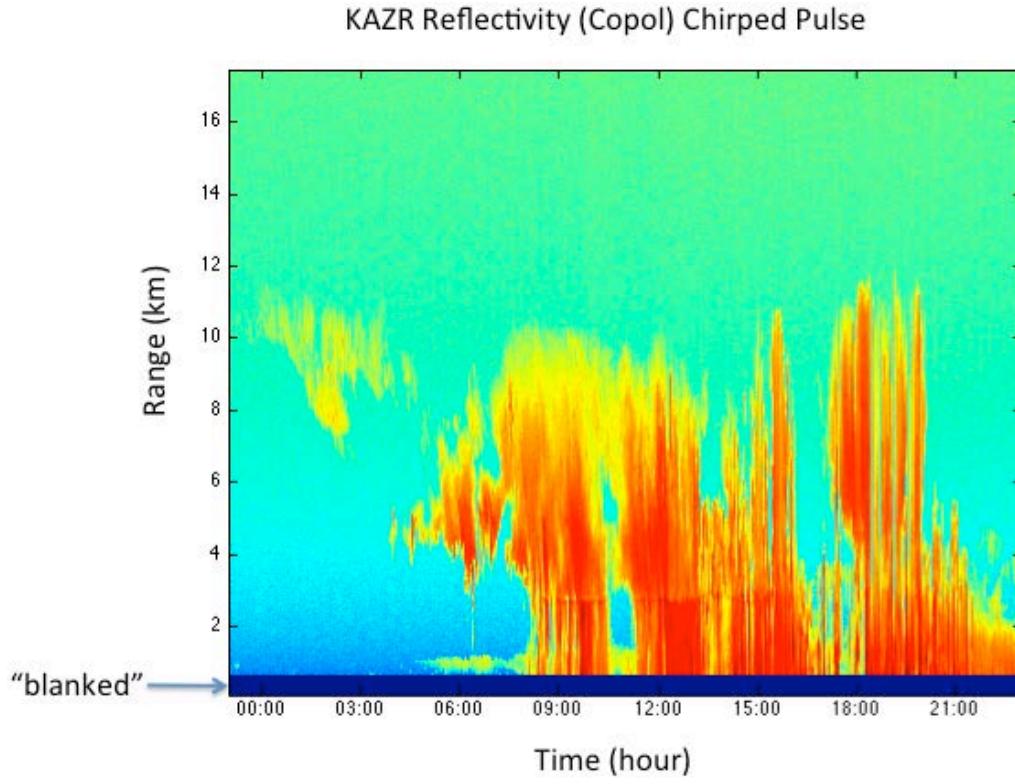
Figure 6. Chirp pulse reflectivity plot.

5.3 User Notes and Known Problems

Please see the ARM Data Quality Reports for the latest list of currently known issues with KAZR data.

5.4 Frequently Asked Questions

1. What is the meteorological radar range equation?

This is the equation to determine the reflectivity and is usually given in decibels of Z or dBZ.

$$Z = 10 \log \left(\frac{1024 \ln(2) \lambda^2 R^2 P_r L_a L_{sys}}{10^{-18} c \tau \pi^3 G_0^2 |K_w|^2 \theta_{3dB}^2 P_t} \right) \quad dBZ$$

where:
  Z = reflectivity (dBZ)
  λ = wavelength (m)
  R = range (m)
  P_r = received power (watts)
  L_a = two-way atmospheric loss
  L_{sys} = radar system losses
  c = speed of light (m/s)
  τ = pulse width (s)
  G_o = antenna gain
  $|K_w|^2$ = index of refraction factor for liquid water at 0 C
  θ_{3dB} = antenna beamwidth
  P_t = transmit power (watts)

2. What index of refraction for water is used to computer reflectivity in the radar range equation $|K_w|^2$?

0.88

3. What is the lifetime of the transmitter and can it be refurbished after it has died?

The advertised lifetime of the Traveling Wave Tube is 20,000 operating hours. This works out to 2.3 years of continuous operation. We track the transmit power output, which provides an indication of when a TWT is about to fail. We have a spare TWT Amplifier (TWTA) for each radar installation to minimize downtime.

Unfortunately, once the TWT has reached the end of its life, it cannot be refurbished. A brand new TWT is installed in the TWTA by the vendor. In addition to spare TWTAs for each installation, we maintain a set of spare TWTs at the SGP warehouse. When there is a failure, we send the failed TWTA along with a new TWT to the vendor for replacement and recertification.

4. How are the burst and chirp pulses transmitted?

Figure 7 shows how we plan on operating the KAZR. Right now, we transmit a chirp pulse immediately followed by a burst pulse. Through the modification of the pulse widths of these pulses, we will be able to optimize the KAZR's sensitivity under varying cloud regimes.
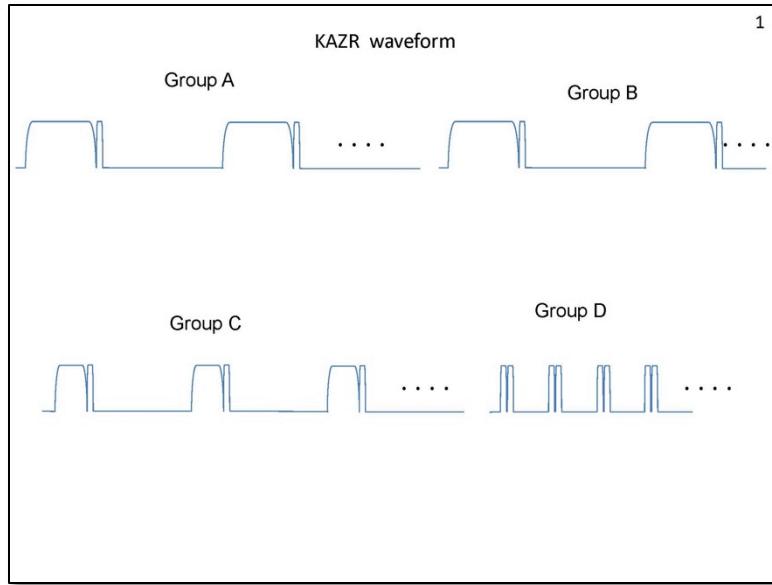
Figure 7. Burst/chirp pulse timing diagram.

6.0   Data Quality

6.1   Data Quality Health and Status

The Data Quality Office (DQO) website has links to several tools for inspecting and assessing KAZR data quality:

- DQ Explorer

- DQ Plot Browser

- NCVweb: Interactive web-based tool for viewing ARM data

Plots of reflectivity, Doppler radial velocity, and Doppler spectral width provide a good indicator of whether the system is operational or not.

The KAZR outputs health and status files that are coincident with the moments and spectra data files. These files are ingested to netCDF. The current format of this health and status file is:

```
dimensions:
    time = UNLIMITED ; // (33119 currently)
variables:
    int base_time ;
        base_time:string = "2-Oct-2011,0:59:58 GMT" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2011-10-02 00:59:58 0:00" ;
```

```
double time(time) ;
    time:long_name = "Time offset from midnight" ;
    time:units = "seconds since 2011-10-02 00:00:00 0:00" ;
int lock_alarm_120(time) ;
    lock_alarm_120:long_name = "Lock alarms for 120 MHz oscillator." ;
    lock_alarm_120:units = "unitless" ;
    lock_alarm_120:missing_value = -9999.f ;
    lock_alarm_120:comment = "0=unlocked, 1=locked" ;
int lock_alarm_360(time) ;
    lock_alarm_360:long_name = "Lock alarms for 360 MHz oscillator." ;
    lock_alarm_360:units = "unitless" ;
    lock_alarm_360:missing_value = -9999.f ;
    lock_alarm_360:comment = "0=unlocked, 1=locked" ;
int lock_alarm_1620(time) ;
    lock_alarm_1620:long_name = "Lock alarms for 1620 MHz oscillator." ;
    lock_alarm_1620:units = "unitless" ;
    lock_alarm_1620:missing_value = -9999.f ;
    lock_alarm_1620:comment = "0=unlocked, 1=locked" ;
int lock_alarm_16560(time) ;
    lock_alarm_16560:long_name = "Lock alarms for 16560 MHz oscillator." ;
    lock_alarm_16560:units = "unitless" ;
    lock_alarm_16560:missing_value = -9999.f ;
    lock_alarm_16560:comment = "0=unlocked, 1=locked" ;
float temp_outside_air(time) ;
    temp_outside_air:long_name = "Outside air temperature" ;
    temp_outside_air:units = "C" ;
    temp_outside_air:missing_value = -9999.f ;
float temp_power_supply(time) ;
    temp_power_supply:long_name = "DC power supply temperature" ;
    temp_power_supply:units = "C" ;
    temp_power_supply:missing_value = -9999.f ;
float temp_antenna_feedhorn(time) ;
    temp_antenna_feedhorn:long_name = "Antenna feedhorn temperature" ;
    temp_antenna_feedhorn:units = "C" ;
    temp_antenna_feedhorn:missing_value = -9999.f ;
float temp_circulator(time) ;
    temp_circulator:long_name = "4-port circulator temperature" ;
    temp_circulator:units = "C" ;
    temp_circulator:missing_value = -9999.f ;
float temp_reverse_power(time) ;
    temp_reverse_power:long_name = "Reverse power termination temperature" ;
    temp_reverse_power:units = "C" ;
    temp_reverse_power:missing_value = -9999.f ;
float temp_rf_detector(time) ;
    temp_rf_detector:long_name = "RF detector temperature" ;
    temp_rf_detector:units = "C" ;
    temp_rf_detector:missing_value = -9999.f ;
float temp_noise_diode(time) ;
    temp_noise_diode:long_name = "Noise diode temperature" ;
    temp_noise_diode:units = "C" ;
    temp_noise_diode:missing_value = -9999.f ;
```

```
float temp_lna(time) ;
    temp_lna:long_name = "LNA temperature" ;
    temp_lna:units = "C" ;
    temp_lna:missing_value = -9999.f ;
float temp_component_plate(time) ;
    temp_component_plate:long_name = "Component plate temperature" ;
    temp_component_plate:units = "C" ;
    temp_component_plate:missing_value = -9999.f ;
float temp_antenna(time) ;
    temp_antenna:long_name = "Antenna temperature" ;
    temp_antenna:units = "C" ;
    temp_antenna:missing_value = -9999.f ;
float temp_local(time) ;
    temp_local:long_name = "Temperature measured by sensor on radar control board" ;
    temp_local:units = "C" ;
    temp_local:missing_value = -9999.f ;
float rh_antenna(time) ;
    rh_antenna:long_name = "Antenna relative humidity" ;
    rh_antenna:units = "%" ;
    rh_antenna:missing_value = -9999.f ;
float rh_local(time) ;
    rh_local:long_name = "Relative humidity measured by sensor on radar control board" ;
    rh_local:units = "%" ;
    rh_local:missing_value = -9999.f ;
float input_minus5V(time) ;
    input_minus5V:long_name = "-5 V power supply voltage" ;
    input_minus5V:units = "volts" ;
    input_minus5V:missing_value = -9999.f ;
float input_5V(time) ;
    input_5V:long_name = "+5 V power supply voltage" ;
    input_5V:units = "volts" ;
    input_5V:missing_value = -9999.f ;
float input_8V(time) ;
    input_8V:long_name = "+8 V power supply voltage" ;
    input_8V:units = "volts" ;
    input_8V:missing_value = -9999.f ;
float input_12V(time) ;
    input_12V:long_name = "+12 V power supply voltage" ;
    input_12V:units = "volts" ;
    input_12V:missing_value = -9999.f ;
float input_15V(time) ;
    input_15V:long_name = "+15 V power supply voltage" ;
    input_15V:units = "volts" ;
    input_15V:missing_value = -9999.f ;
float power_supply_5V(time) ;
    power_supply_5V:long_name = "+5 V power supply voltage on radar control board" ;
    power_supply_5V:units = "volts" ;
    power_supply_5V:missing_value = -9999.f ;
float power_supply_5p2V(time) ;
    power_supply_5p2V:long_name = "+5.2 V power supply voltage on radar control board" ;
    power_supply_5p2V:units = "volts" ;
```

```
        power_supply_5p2V:missing_value = -9999.f ;
float power_supply_28V(time) ;
        power_supply_28V:long_name = "+28 V power supply voltage on radar control board" ;
        power_supply_28V:units = "volts" ;
        power_supply_28V:missing_value = -9999.f ;
int twta_status_state(time) ;
        twta_status_state:long_name = "TWTA status state" ;
        twta_status_state:units = "unitless" ;
        twta_status_state:missing_value = -9999.f ;
int is_powered_on(time) ;
        is_powered_on:long_name = "On/Off state" ;
        is_powered_on:units = "unitless" ;
        is_powered_on:missing_value = -9999.f ;
int remote_control_enabled(time) ;
        remote_control_enabled:long_name = "Remote control enabled (0/1)" ;
        remote_control_enabled:units = "unitless" ;
        remote_control_enabled:missing_value = -9999.f ;
int fault_flags(time) ;
        fault_flags:long_name = "Instrument fault flags ... need a list of possible values" ;
        fault_flags:units = "unitless" ;
        fault_flags:missing_value = -9999.f ;
int pdu_status_j1_logic(time) ;
        pdu_status_j1_logic:long_name = "PDU status j1 logic ... need description" ;
        pdu_status_j1_logic:units = "unitless" ;
        pdu_status_j1_logic:missing_value = -9999.f ;
int j2_logic(time) ;
        j2_logic:long_name = "j2 logic ... need description" ;
        j2_logic:units = "unitless" ;
        j2_logic:missing_value = -9999.f ;
double spare_analog_j12_1(time) ;
        spare_analog_j12_1:long_name = "spare analog ... need a list of possible values" ;
        spare_analog_j12_1:units = "unitless" ;
        spare_analog_j12_1:missing_value = -9999.f ;
float cal_constant_ch1_copol(time) ;
        cal_constant_ch1_copol:long_name = "Radar calibration constant, channel 1, copolar" ;
        cal_constant_ch1_copol:units = "dB" ;
        cal_constant_ch1_copol:missing_value = -9999.f ;
        cal_constant_ch1_copol:comment = "Values are updated on an approximately hourly basis, as
        new raw data files are collected." ;
float cal_constant_ch1_xpol(time) ;
        cal_constant_ch1_xpol:long_name = "Radar calibration constant, channel 1, cross-polar" ;
        cal_constant_ch1_xpol:units = "dB" ;
        cal_constant_ch1_xpol:missing_value = -9999.f ;
        cal_constant_ch1_xpol:comment = "Values are updated on an approximately hourly basis, as new
        raw data files are collected." ;
float cal_constant_ch2_copol(time) ;
        cal_constant_ch2_copol:long_name = "Radar calibration constant, channel 2, copolar" ;
        cal_constant_ch2_copol:units = "dB" ;
        cal_constant_ch2_copol:missing_value = -9999.f ;
        cal_constant_ch2_copol:comment = "Values are updated on an approximately hourly basis, as
        new raw data files are collected." ;
```

float cal_constant_ch2_xpol(time) ;
    cal_constant_ch2_xpol:long_name = "Radar calibration constant, channel 2, cross-polar" ;
    cal_constant_ch2_xpol:units = "dB" ;
    cal_constant_ch2_xpol:missing_value = -9999.f ;
    cal_constant_ch2_xpol:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
float noise_source_ch1(time) ;
    noise_source_ch1:long_name = "Noise source, channel 1" ;
    noise_source_ch1:units = "dBm" ;
    noise_source_ch1:missing_value = -9999.f ;
    noise_source_ch1:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
float noise_source_ch2(time) ;
    noise_source_ch2:long_name = "Noise source, channel 2" ;
    noise_source_ch2:units = "dBm" ;
    noise_source_ch2:missing_value = -9999.f ;
    noise_source_ch2:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
float rx_gain_ch1(time) ;
    rx_gain_ch1:long_name = "Receiver gain, channel 1" ;
    rx_gain_ch1:units = "dB" ;
    rx_gain_ch1:missing_value = -9999.f ;
    rx_gain_ch1:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
float rx_gain_ch2(time) ;
    rx_gain_ch2:long_name = "Receiver gain, channel 2" ;
    rx_gain_ch2:units = "dB" ;
    rx_gain_ch2:missing_value = -9999.f ;
    rx_gain_ch2:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
float rx_noise_ch1(time) ;
    rx_noise_ch1:long_name = "Receiver noise, channel 1" ;
    rx_noise_ch1:units = "dBm" ;
    rx_noise_ch1:missing_value = -9999.f ;
    rx_noise_ch1:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
float rx_noise_ch2(time) ;
    rx_noise_ch2:long_name = "Receiver noise, channel 2" ;
    rx_noise_ch2:units = "dBm" ;
    rx_noise_ch2:missing_value = -9999.f ;
    rx_noise_ch2:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
float tx_power(time) ;
    tx_power:long_name = "Transmit power" ;
    tx_power:units = "dBm" ;
    tx_power:missing_value = -9999.f ;
    tx_power:comment = "Values are updated on an approximately hourly basis, as new raw data files are collected." ;
int fft_taper_type ;
    fft_taper_type:long_name = "Taper factor added to fast fourier transform." ;
    fft_taper_type:units = "unitless" ;

```
    fft_taper_type:missing_value = -9999.f ;
    fft_taper_type:comment = "-1=WINDOW_INVALID, 0=WINDOW_BARTLETT,
    1=WINDOW_BLACKMAN, 2=WINDOW_HAMMING, 3=WINDOW_HANNING,
    4=WINDOW_RECTANGLE, 5=WINDOW_TUKEY" ;
int processing_mode ;
    processing_mode:long_name = "KAZR processing mode" ;
    processing_mode:units = "unitless" ;
    processing_mode:missing_value = -9999.f ;
    processing_mode:comment = "1=pulse pair, 2=FFT mode" ;
int timing_mode ;
    timing_mode:long_name = "KAZR timing mode" ;
    timing_mode:units = "unitless" ;
    timing_mode:missing_value = -9999.f ;
    timing_mode:comment = "1=normal operation for co- and cross polarized sampling,
    2=Interleaved second group of samples for Precip mode, 3=internal calibration mode, 4=noise
    diode on/off for measuring gain and noise figure, 5=TBD" ;
float lat ;
    lat:long_name = "North latitude" ;
    lat:units = "degree_N" ;
    lat:valid_min = -90.f ;
    lat:valid_max = 90.f ;
float lon ;
    lon:long_name = "East longitude" ;
    lon:units = "degree_E" ;
    lon:valid_min = -180.f ;
    lon:valid_max = 180.f ;
float alt ;
    alt:long_name = "Altitude above mean sea level" ;
    alt:units = "m" ;
```

6.2 Data Reviews by Instrument Mentor

Instrument mentors review KAZR data in the following ways:

- Routine review for nominal operation, usually daily Monday–Friday

- When requested by site operations

- When requested by the site scientist team

- When requested by an ARM data translator

- When requested by a data user

- When notified automatically by the KAZR's built-in test (BIT) email messages.

6.3 Data Assessments by Site Scientist/Data Quality Office

To be determined.

6.4 Value-Added Product

The KAZR is a primary input datastream into the Active Remote Sensing of Clouds (ARSCL) value-added product (VAP). See http://www.arm.gov/data/vaps/arscl for more information.

7.0   Instrument Details

See "KAZR Operations Manual."

7.1   Detailed Description

Figure 8 shows a block diagram of the KAZR.
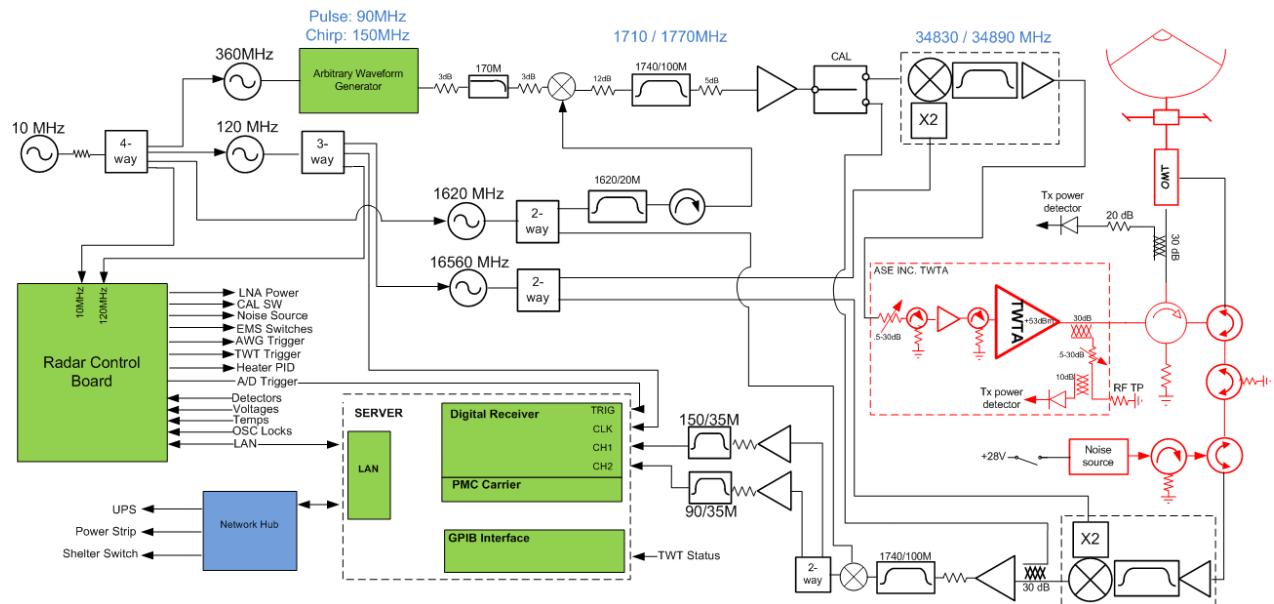
Figure 8. KAZR block diagram.

7.2   Theory of Operation

See "KAZR Operations Manual."

7.3   Calibration

See "KAZR Operations Manual" and ARM Calibration database.

7.4   Operation and Maintenance

See "KAZR Operations Manual."

7.5 Citable References

Mead, J. 2010. MMCR Calibration Study. U.S. Department of Energy. DOE/SC-ARM/TR-088.

Baldi, C, and J Mead. "Ka-Band ARM Zenith Radar (KAZR) System Description and Operations Manual." In progress.

Doviak, RJ, and DS Zrnic. 1993. "Doppler Radar and Weather Observations. 2nd Edition, Academic Press.

Bringi, VN and V Chandrasekar. 2001. "Polarimetric Doppler Weather Radar." Cambridge University Press.