```
AcousticCalibrations
                    AcousticCalibratorsPeriodicTester
                                                                                               AcqueticCalibrator
    + measurementProgress: pygtSignal «class attribute»
                                                                            # info: dict «instance attribute»
    + calibrationProgress: pyqtSignal «class attribute»
                                                                            # cl: int «instance attribute»
    + invalidMeasurementValue: pygtSignal «class attribute»
                                                                            # nominal values: pd.DataFrame «instance attribute»
    + realTimeValues: pygtSignal «class attribute»
                                                                            # pressure_influence: float «instance attribute»
    # acceptanceLimits: pd.DataFrame «class attribute»
                                                                            # free field difference: float «instance attribute»
    # resource_manager: pyVisa.ResourceManager «instance attribute»
                                                                            # calibration_results: pd.DataFrame «instance attribute»
    # DMM: pvVisa.GPIBInstrument «instance attribute»
    # AA: pyVisa.GPIBInstrument «instance attribute»
                                                                            + get_ambient_correction(pressure:
    # standard: AcousticCalibrator «instance attribute»
                                                                            float) → float «instance method»
    # microphone: dict «instance attribute»
                                                                            + set calibration results(level results: np.ndarray,
    # calibration consecutive: str «instance attribute»
                                                                            freq_results: np.ndarray, thd_results: np.ndarray)
    # dut: AcousticCalibrator «instance attribute»
                                                                            → None «instance method»
    # customer info: dict «instance attribute»
    # ambient_condition_values: dict «instance attribute»
    # standard measurement values: list «instance attribute»
    # standard noise values: pd.DataFrame «instance attribute»
    # dut_measurement_values: list «instance attribute»
    # dut noise values: pd.DataFrame «instance attribute»
    # expanded_uncertainty: dict «instance attribute»
    # current level: str «instance attribute»
    + run_main_sequence() → None «instance method»
    + check_state() → None «instance method»
    + set current level(level: int) → None «instance method»
    + set standard calibrator(standard:
    AcousticCalibrator) → None «instance method»
    + set_dut(dut: AcousticCalibrator) → None «instance method»
    + set_customer_info(customer_info: dict) → None «instance method»
    + set_consecutive(consecutive: str) → None «instance method»
    + set_standards(DMM:dict, AA: dict) → None «instance method»
                                                                           «instantiate»
    + measure_mic_sensibility() → None «instance method»
+ measure_ambient_noise() → float «instance method»
    + measure_quantities(n_indications:
                                                                                                      GUIController
    int=30) → None «instance method»
    + estimate expanded uncertainty() → None «instance method»
                                                                           # bruel4231: AcousticCalibrator «class attribute»
    + report_ambient_conditions(temperature:float,
                                                                           # cal21: AcousticCalibrator «class attribute»
    humidity:float, pressure: float) → None «instance method»
                                                                           # grass40CE: ditc «class attribute»
    + set_stage(stage: int) → None «instance method» + set state(state: int) → None «instance method»
                                                                           # TESTER: AcousticCalibratorsPeriodicTester «instance attribute»
                                                                           # qui: AcousticCalibratorsUI «instance attribute»
    + volt2level(v:float, ref_lev: tuple) → float «static
                                                                           # resource_manager: pyVisa.ResourceManager «instance attribute»
    method»
                                                                           + standard_calibrator: AcousticCalibrator «instance attribute»
                                                                           + acquired_values: pd.DataFrame «instance attribute»
                                                                           + standard averaged values: pd.DataFrame «instance attribute»
                                                                           + dut_averaged_values: pd.DataFrame «instance attribute»
                                                    «instantiate»
Q0bject
                                                                           + save standards state: bool «instance attribute»
                                                                           + save_DUT_info_state: bool «instance attribute»
     DX
                                                            «instantiate» + self_test_passed: bool «instance attribute»
                                                                           + searchStandardWorker: PyQt5.QObject «instance attribute»
                    StandardsSearcher
                                                                           + selfTesterWorker: PyQt5.QObject «instance attribute»
+ finished: pygtSignal «class attribute»
                                                                           + calibrationThread: PyOt5.OThread «instance attribute»
+ progress: pyqtSignal «class attribute»
# resources_manager: visa.ResourceManager
                                                                           # connect_signals() → None «instance method»
«instance attribute »
                                                                           + fill mic info() → None «instance method»
+ DMM model: str «instance attribute»
                                                                           + fill_standard_calibrator_info() → None «instance method»
+ AA_model: str «instance attribute»
                                                                           + save_standards_info() → None «instance method»
+ resources: tuple «instance attribute »
                                                                           + save_dut_info() → None «instance method»
+ data: dict «instance attribute »
                                                                           + self_test() → None «instance method»
                                                                           + show_self_test_results(results: tuple) → None «instance method»
+ search() → None «instance method »
                                                                           + search_standards() → None «instance method»
                                                                           + fill_standards_info() → None «instance method»
                                                            «instantiate»
                                                                          + update_noise_values() → None «instance method»
                        SelfTester
                                                                           + show_standard_noise_values() → None «instance method»
                                                                           + show dut noise values() → None «instance method»
+ finished: pyqtSignal «class attribute»
                                                                           + show_averaged_standard_values() → None «instance method»
+ progress: pygtSignal «class attribute»
                                                                           + show_averaged_dut_values() → None «instance method»
# DMM: visa.Resource «instance attribute»
                                                                           + update_real_time_values() → None «instance method»
# AA: visa.Resource «instance attribute»
                                                                           + start() → None «instance method»
                                                                           + pause() → None «instance method»
+ self test() → None «instance method»
                                                                           + sequence_control() → None «instance method»
                                                                           + eval_answer(instruction:
                                                            «instantiate»
                                                                          PyQt5.QMessageBox) → None «instance method»
                  AcousticCalibratorsUI
                                                                           + invalid_measurement(value: float) → None «instance
                                                                           method»
«some instance attributes corresponding to the GUI
elements»
                                                                                  QMainWindow
«a few instance methods for configuring the initial state
of the GUI elements»
```