

```

344             i += 1
345
346         # Finally, Check for Step Information
347         if "stepped" in self.raw_params["Flags"]:
348             self._load_step_information(raw_filename)
349
350     def get_raw_property(self, property_name=None):
351         """Get a property. By default it returns
    everything"""
352         if property_name is None:
353             return self.raw_params
354         elif property_name in self.raw_params.keys():
355             return self.raw_params[property_name]
356         else:
357             return "Invalid property. Use %s" % str(self.
    raw_params.keys())
358
359     def get_trace_names(self):
360         return [trace.name for trace in self._traces]
361
362     def get_trace(self, trace_ref):
363         """Retrieves the trace with the name given. """
364         if isinstance(trace_ref, str):
365             for trace in self._traces:
366                 if trace_ref == trace.name:
367                     # assert isinstance(trace, DataSet)
368                     return trace
369             return None
370         else:
371             return self._traces[trace_ref]
372
373     def _load_step_information(self, filename):
374         # Find the extension of the file
375         if not filename.endswith(".raw"):
376             raise LTSpiceReadException("Invalid Filename
    . The file should end with '.raw'")
377         logfile = filename[:-3] + 'log'
378         try:
379             log = open(logfile, 'r')
380         except:
381             raise LTSpiceReadException("Step information
    needs the '.log' file generated by LTSpice")
382
383         for line in log:
384             if line.startswith(".step"):

```