```
module ieee802-ethernet-pse{
  yang-version 1.1;
  namespace "urn:ieee:std:802.3:yang:ieee802-ethernet-pse";
  prefix pse;

  import ietf-interfaces {
    prefix "if";
    reference "IETF RFC 7223";
  }

  import ieee802-ethernet-interface {
    prefix eth-if;
  }

  import ietf-yang-types {
    prefix yang;
    reference "IETF RFC 6991";
  }

  organization
    "IEEE 802.3 Ethernet Working Group
     Web URL: http://www.ieee802.org/3/";

  contact
    "Web URL: http://www.ieee802.org/3/cf/";

  description
    "This module contains YANG definitions for configuring and
     managing ports with Power Over Ethernet feature defined by
     IEEE 802.3. It provides functionality roughly equivalent to
     that of the POWER-ETHERNET-MIB defined in IETF RFC 3621.";

  reference "IEEE Std 802.3-2018, unless dated explicitly";

  typedef multi-pair-detection-state {
    type enumeration {
      enum disabled {
        value 1;
        description "PSE disabled.";
      }
      enum searching {
        value 2;
        description "PSE is searching.";
      }
      enum deliveringPower {
        value 3;
        description "PSE is delivering power.";
      }
      enum fault {
        value 4;
        description "PSE fault detected.";
      }
      enum test {
        value 5;
        description "PSE test mode.";
      }
      enum otherFault {
```

Deleted:

Deleted: .

```
        value 6;
        description "PSE implementation specific fault detected.";
      }
    }
    description
      "Detection state of a multi-pair PSE.";
    reference
      "IEEE Std 802.3, 30.9.1.1.5";
  }

  typedef single-pair-detection-state{
    type enumeration {
      enum unknown {
        value 1;
        description "True detection state unknown.";
      }
      enum disabled {
        value 2;
        description "PoDL PSE is disabled.";
      }
      enum searching {
        value 3;
        description "PoDL PSE is searching.";
      }
      enum deliveringPower {
        value 4;
        description "PoDL PSE is delivering power.";
      }
      enum sleep {
        value 5;
        description "PoDL PSE is in sleep state.";
      }
      enum idle {
        value 6;
        description "PoDL PSE is idle.";
      }
      enum error {
        value 7;
        description "PoDL PSE error.";
      }
    }

    description
      "Detection state of a PoDL PSE.";
    reference
      "IEEE Std 802.3, 30.15.1.3";
  }

  typedef power-class {
    type enumeration {
      enum class0 {
        value 1;
        description "Class 0";
      }
      enum class1 {
        value 2;
        description "Class 1";
```

Deleted: d

Deleted: t

Deleted: d

Deleted: c

Deleted: c

```
        }
        enum class2 {
          value 3;
          description "Class 2";
        }
        enum class3 {
          value 4;
          description "Class 3";
        }
        enum class4 {
          value 5;
          description "Class 4";
        }
        enum class5 {
          value 6;
          description "Class 5 (for PoDL-only)";
        }
        enum class6 {
          value 7;
          description "Class 6 (for PoDL-only)";
        }
        enum class7 {
          value 8;
          description "Class 7 (for PoDL-only)";
        }
        enum class8 {
          value 9;
          description "Class 8 (for PoDL-only)";
        }
        enum class9 {
          value 10;
          description "Class 9 (for PoDL-only)";
        }
        enum unknown {
          value 11;
          description "Initializing, true state not yet known (only for PoDL
PSE).";
        }
      }
      description
        "Power Class";
      reference
        "IEEE Std 802.3, 30.9.1.1.6 aPSEPowerClassification and
         IEEE Std 802.3, 30.15.1.1.6 aPoDLPSEDetectedPDPowerClass.";
    }

    identity pse-type {
      description "Base type for PSE.";
    }

    identity multi-pair {
      base pse-type;
      description "PSE supports IEEE Std 802.3, Clause 33.";
    }

    identity single-pair {
      base pse-type;
```

Deleted: c
Deleted: c
Deleted: c
Deleted: c
Deleted: c
Deleted: c
Deleted: c
Deleted:
Deleted: .
Deleted: .
Deleted: i
Deleted: .
Deleted: p
Deleted: c
Deleted:
Deleted: .
Deleted:

```
    description "PSE support IEEE Std 802.3, Clause 104.";
  }

  identity powering-pairs {
    description "Base type for powering pairs.";
  }
  identity signal {
    base powering-pairs;
    description "The signal pair is in use.";
  }
  identity spare {
    base powering-pairs;
    description "The spare pair is in use.";
  }

  augment "/if:interfaces/if:interface/eth-if:ethernet" {
    description
      "Augments ethernet interface configuration model with
       nodes specific to DTE Power via MDI devices and ports.";

    container pse {
      description
        "DTE Power via MDI port configuration.";
      reference
        "IEEE Std 802.3, 30.9.1 PoE PSE & IEEE Std 802.3, 30.15.1 PoDL
         PSE";

      leaf supported-pse-type {
        type identityref {
          base pse:pse-type ;
        }
        config false;
        description
          "PSE may support IEEE Std 802.3, Clause 33 or IEEE Std 802.3,
Clause 104.";
      }

      container multi-pair {
        presence "PSE port supports IEEE Std 802.3, Clause 33.";

        description
          "PSE port configuration in IEEE Std 802.3, 30.9.1.";

        leaf pse-enable {
          type boolean;
          default false;
          description
            "Whether to enable the PSE function on the interface.";
          reference
            "IEEE Std 802.3, 30.9.1.1.2 aPSEAdminState";
        }

        leaf powering-pairs {
          type identityref{
            base powering-pairs;
          }
          description
```

Deleted: S

Deleted:

Deleted: .

Deleted:

Deleted: .

Deleted:

Deleted: e

Deleted: w

Deleted: . .

Deleted: .

```
              "Describes or controls the PSE pairs in use. If the value of
               pethPsePortPowerPairsControl is true, this object is
    writeable.";

            reference
              "IEEE Std 802.3, 30.9.1.1.4 aPSEPowerPairs";
          }

          leaf pairs-control-ability {
            type boolean;
            default true;
            config false;
            description
              "Describes the capability of controlling the power pairs
               functionality to switch pins for sourcing power.";
            reference
              "IEEE Std 802.3, 30.9.1.1.3 aPSEPowerPairsControlAbility";
          }

          leaf detection-status {
            type multi-pair-detection-state;
            config false;
            description
              "Describes the operational status of the port
               PD detection.";
            reference
              "IEEE Std 802.3, 30.9.1.1.5 aPSEPowerDetectionStatus";
          }

          leaf classifications {
            when "../detection-status = 'deliveringPower'" {
              description
                "This node only applies when the detection status is
                 delivering power.";
            }
            type power-class;
            config false;
            description "The power class of the port.";
            reference
              "IEEE Std 802.3, 30.9.1.1.6 aPSEPowerClassfication";
          }

          container statistics {
            config false;
            description "Statistics information of the multi-pair port.";

            leaf power-denied {
              type yang:counter64;
              description
                "This counter is incremented when the PSE state diagram
                 enters the state POWER_DENIED.";
              reference
                "IEEE Std 802.3, 30.9.1.1.8 aPSEPowerDeniedCounter";
            }

            leaf invalid-signature {
              type yang:counter64;
```

Deleted: y

Deleted: p

Deleted: s

```
      description
        "This counter is incremented when the PSE state diagram
         enters the state SIGNATURE_INVALID.";
      reference
        "IEEE Std 802.3, 30.9.1.1.7 aPSEInvalidSignatureCounter";
    }

    leaf mps-absent {
      type yang:counter64;
      description
        "This counter is incremented when the PSE state diagram
         transitions directly from the state POWER_ON to the
         state IDLE due to tmpdo_timer_done being asserted.";
      reference
        "IEEE Std 802.3, 30.9.1.1.11 aPSEMPSAbsentCounter";
    }

    leaf overload {
      type yang:counter64;
      description
        "This counter is incremented when the PSE state diagram
         enters the state ERROR_DELAY_OVER.";
      reference
        "IEEE Std 802.3, 30.9.1.1.9 aPSEOverLoadCounter";
    }

    leaf short {
      type yang:counter64;
      description
        "This counter is incremented when the PSE state diagram
         enters the state ERROR_DELAY_SHORT, per IEEE Std 802.3,
         Figure 33-9.";
      reference
        "IEEE Std 802.3, 30.9.1.1.10 aPSEShortCounter";
    }

    leaf cumulative-energy {
      type yang:counter64;
      units 'millijoule';
      description
        "The cumulative energy supplied by the PSE as measured at the
         MDI in millijoules.";

      reference
        "IEEE Std 802.3, 30.9.1.1.14 aPSECumulativeEnergy";
    }

  }

  leaf actual-power {
    type decimal64 {
      fraction-digits 4 ;
    }

    units 'milliwatt';
    config false;
    description
```

Deleted:

Deleted:

```
        "The actual power drawn by a PD over the port.";
      reference
        "IEEE Std 802.3, 30.9.1.1.12 aPSEActualPower";
    }

    leaf power-accuracy {
      type uint64;
      units 'milliwatt';
      config false;
      description
        "An integer value indicating the accuracy
         associated with aPSEActualPower in +/- milliwatts.";
      reference
        "IEEE Std 802.3, 30.9.1.1.13 aPSEPowerAccuracy";
    }

  }

  container single-pair {
    presence "PSE port working in PoDL.";

    description
      "PoDL PSE configuration as defined in IEEE Std 802.3, 30.15.1.";

    leaf pse-enable {
      type boolean;
      default false;
      description
        "whether to enable the PSE function on the interface.";
      reference
        "IEEE Std 802.3, 30.15.1.1.2 aPoDLPSEAdminState";
    }

    leaf detection-status {
      type single-pair-detection-state;
      config false;
      description
        "Indicates the current status of the PoDL PSE.";
      reference
        "IEEE Std 802.3, 30.15.1.1.3 aPoDLPSEPowerDetectionStatus";
    }

    leaf podl-type {
      type enumeration {
        enum unknown {
          description "Unknown PSE type.";
        }
        enum typeA {
          description "Type A";
        }
        enum typeB {
          description "Type B";
        }
        enum typeC {
          description "Type C";
        }
        enum typeD {
```

Deleted: t

Deleted:

Deleted: u

Deleted: t

Deleted: t

Deleted: t

```
        description "Type_D";
      }
    }
    config false;
    description "PSE type specified in IEEE Std 802.3, 104.4.1.";
  }

  leaf detected-pd-type {
    when "../detection-status = 'deliveringPower'" {
      description
        "This node only applies when the detection status is
         delivering power.";
    }

    type enumeration {
      enum unknown {
        description "Unknown";
      }
      enum typeA {
        description "Type_A";
      }
      enum typeB {
        description "Type_B";
      }
      enum typeC {
        description "Type_C";
      }
      enum typeD {
        description "Type_D";
      }
    }
    config false;
    description
      "Indicates the Type of the detected PoDL PD as specified in IEEE
       Std 802.3, 104.5.1.";

    reference
      "IEEE Std 802.3, 30.15.1.1.5 aPoDLPSEDetectedPDType";
  }

  leaf pd-power-class {

    when "../detection-status = 'deliveringPower'" {
      description
        "This node only applies when the detection status is
         delivering power.";
    }

    type power-class;
    config false;
    description
      "Power class of the port.";
    reference
      "IEEE Std 802.3, 30.15.1.1.6 aPoDLPSEDetectedPDPowerClass";
  }

  container statistics {
```

Deleted: t

Deleted: y

Deleted: u

Deleted: t

Deleted: t

Deleted: t

Deleted: t

Deleted: i

Deleted: y

Deleted: p

```
config false;
description "Statistics information of the single-pair PSE.";

leaf power-denied {
  type yang:counter64;
  description
    "This counter is incremented when the PoDL PSE state diagram
     variable power_available transitions from true to false (see
     IEEE Std 802.3, 104.4.3.3).";
  reference
    "IEEE Std 802.3, 30.15.1.1.9 aPoDLPSEPowerDeniedCounter";
}

leaf invalid-signature {
  type yang:counter64;
  description
    "This counter is incremented when the PSE state diagram
     enters the state SIGNATURE_INVALID.";
  reference
    "IEEE Std 802.3, 30.15.1.1.7 aPoDLPSEInvalidSignatureCounter";
}

leaf invalid-class {
  type yang:counter64;
  description
    "This counter is incremented when the PoDL PSE state diagram
     variable tclass_timer_done transitions from false to true or
     when the valid_class variable transitions from true to false
     (see IEEE Std 802.3, 104.4.3.3).";

  reference
    "IEEE Std 802.3, 30.15.1.1.8 aPoDLPSEInvalidClassCounter";
}

leaf overload {
  type yang:counter64;
  description
    "This counter is incremented when the PSE state diagram
     variable overload_held transitions from false to true (see
     IEEE Std 802.3, 104.4.3.3).";
  reference
    "IEEE Std 802.3, 30.15.1.1.10 aPoDLPSEOverLoadCounter";
}

leaf fvs-absence {
  type yang:counter64;
  description
    "Maintain Full Voltage Signature absent counter.
     This counter is incremented when the PoDL PSE state diagram
     variable mfvs_timeout transitions from false to true (see
     IEEE Std 802.3, 104.4.3.3).";

  reference
    "IEEE Std 802.3, 30.15.1.1.11
     aPoDLPSEMaintainFullVoltageSignatureAbsentCounter";
}
```

Deleted: s

```
          leaf cumulative-energy {
            type yang:counter64;
            description
              "A count of the cumulative energy supplied by the PoDL PSE,
               measured at the MDI, and expressed in units of millijoules.";
            reference
              "IEEE Std 802.3, 30.15.1.1.14 aPoDLPSECumulativeEnergy";
          }

        }

        leaf actual-power {
          type decimal64 {
            fraction-digits 4 ;
          }

          units 'watt';
          config false;
          description
            "An integer value indicating present (actual) power being
             supplied by the PoDL PSE as measured at the MDI in
             milliwatts.";
          reference
            "IEEE Std 802.3, 30.15.1.1.12 aPoDLPSEActualPower";
        }

        leaf power-accuracy {
          type uint64;
          units 'milliwatt';
          config false;
          description
            "A signed integer value indicating the accuracy associated with
             aPoDLPSEActualPower in milliwatts.";
          reference
            "IEEE Std 802.3, 30.15.1.1.13 aPoDLPSEPowerAccuracy";
        }
      }
    }
  }
}
```