### Gestione Essiccatore

#### Descrizione

La gestione essiccatore si occupa di visualizzare e gestire il funzionamento del tamburo essiccatore o dei tamburi essiccatori dove è prevista la presenza di due tamburi (ParallelDrum)

#### Parametri

##### ParaTabBruc\_ReadFile()

Lettura parametri su init applicazione, chiamata da ParametriReadFile(), chiamata da Form\_Load().

##### ParaTabBruc\_Apply()

Aggiornamento interfaccia da lettura parametri , chamata da ParametriApply().

#### Logica

##### GestSetRegBruciatore

##### GestioneFunzAutomaticoBruc

Controllo di retroazione su elemento di tipo modulatore (up/down)

La posizione del modulatore del bruciatore è funzione della temperatura che si vuole ottenere (impostata da parametri), il calcolo della posizione è però dipendente dalla portata del materiale immesso, la quale è però dipendente dalla temperatura (vedi GestSetRegBruciatore()). Calcolata la portata teorica in dipendenza della temperatura si utilizza una curva parametrica di relazione tra apertura bruciatore (%) portata predosatori (%), sulla base della portata si stabilisce la apertura del bruciatore da raggiungere. Il controllo applicato è una retroazione con finestra di isteresi del 5% (vedi GestTotRegBruciatore()): nel caso in cui la differenza tra valore richiesto e valore rilevato sia maggiore del 5% viene aperto il modulatore (up), nel caso in cui la differenza **assoluta** tra valore rilevato e valore richiesto sia minore del 5% e superiore dello 0.5% viene attivato un controllo a impulsi di apertura del modulatore di durata minima di 500 ms e massima di 2000ms con ritardo tra impulsi di 3 secondi.

##### RegolazioneAriaAspiratore

Regolazione del ModulatoreFumiTamburo.

##### Moduli interessati alla gestione essiccatore

BrucAuto.bas

<descrizione>

ControlloBruc.bas

<descrizione>

ParaTabBruc.bas

<descrizione>

#### Metodi utilizzati nella gestione essiccatore:

BrucAuto.bas

306 - Private Sub GestSetRegBruciatore()

291 - Public Sub GestSetRegBruciatore2()

116 - Private Sub GestTotRegBruciatore(tamburo As Integer)

58 - Public Sub GestRegolazioneBruciatore(tamburo As Integer)

12 - Public Sub AttivaUscitePerRegolazioneEss(verso As Integer, smSec As Long, tamburo As Integer)

8 - Private Sub TableShift(ByRef lista() As Double, listaDim As Integer, verso As Integer)

ControlloBruc.bas

205 - Public Sub RegolazioneAriaAspiratore()

148 - Public Sub FiammaBruciatorePresente\_change(tamburo As Integer)

123 - Public Function VerificaCondizioniAvviamentoBruciatore(tamburo As Integer, aCaldo As Boolean) As Long

106 - Public Sub ConteggioTempoArrestoBruciatore(tamburo As Integer)

102 - Public Sub StopBruciatoreTamburo(tamburo As Integer)

78 - Public Sub ControlliFiammaBruciatore(tamburo As Integer)

70 - Public Sub ControlloBloccoBruciatore(tamburo As Integer)

69 - Public Sub AllarmeCombustibile(tamburo As Integer, inizializza As Boolean)

59 - Public Sub ModulatoreBruciatore\_change(tamburo As Integer)

59 - Public Sub GestioneFunzAutomaticoBruc(tamburo As Integer)

56 - Public Sub ModoRegolazioneDepressioneFiltroConDepressFiltroIN()

40 - Public Sub RegolazioneAriaFredda()

40 - Public Sub AltaTemperaturaFumiTamburo(tamburo As Integer)

40 - Public Sub AltaTemperaturaFiltroSw()

39 - Public Sub GestioneModulatoreBruc(tamburo As Integer)

38 - Public Sub ValoreLettoTempScivolo\_change(tamburo As Integer)

37 - Public Sub RegolazioneImpulsoAriaFredda(UpDown As Integer, DeltaTemperatura As Integer)

36 - Public Sub TempTorre\_change(Index As Integer, temperatura As Long)

35 - Public Sub ArrestoBrucTempoX(tamburo As Integer)

33 - Public Sub LetturaScivoloTamburo(tamburo As Integer, temperaturaScivolo As Long, plcInAnalogici\_Fatta As Boolean)

30 - Public Sub BloccoFiammaBruciatore\_change(tamburo As Integer)

29 - Public Sub StopBruciatore(tamburo As Integer)

29 - Public Sub AggiornaGraficaValvolaCombustibile\_Change()

26 - Public Sub TempUscitaFiltro\_change()

25 - Public Sub TempFumiTamburo\_change(tamburo As Integer)

24 - Public Sub LetturaDepressioneBruciatore(tamburo As Integer, depressioneBruciatore As Long, plcInAnalogici\_Fatta As Boolean)

24 - Public Sub ArrestoBrucITT()

23 - Public Sub AllarmePerditaValvoleBrucOC\_change(tamburo As Integer)

22 - Public Sub SicurezzaTempOlioComb\_change(tamburo As Integer)

22 - Public Sub ModulatoreAspFiltro\_change()

22 - Public Sub AllarmePressioneBrucAlta\_change(tamburo As Integer)

22 - Public Sub AllarmePerditaValvoleBruc\_change(tamburo As Integer)

22 - Public Sub AggiornaTemperaturaTorre()

21 - Public Function BrucInAccensione(tamburo As Integer) As Boolean

20 - Public Sub StartBruciatore(tamburo As Integer)

20 - Public Sub ModulatoreFumiTamburo\_change(tamburo As Integer)

20 - Public Sub CompressoreBruciatorePressioneInsuff\_change()

19 - Public Sub TempEntrataFiltro\_change()

19 - Public Sub DeflettoreAntincendioTamburo\_change(tamburo As Integer)

19 - Public Sub BruciatoreModulatore\_change(tamburo As Integer)

18 - Public Sub BruciatoreInManuale(tamburo As Integer)

18 - Public Function DurataImpulsoAriaFredda(DeltaTemperatura As Integer) As Integer

16 - Public Sub TempSottoMesc\_change()

16 - Public Sub PortaModASetAvvioCaldo(tamburo As Integer)

16 - Public Sub ImpulsiContalitriCombustibile\_change(tamburo As Integer)

15 - Public Sub ModoRegolazioneDepressioneFiltroConDepressTamburo()

15 - Public Function PortaModulatoreASet(target As Long, letturaposizione As Long, Tolleranza As Integer, ByRef fatto As Boolean) As ModulatoreStatusEnum

14 - Public Sub ValoreLettoDepressioneBruc\_change(tamburo As Integer)

14 - Public Sub TempSondaAggiuntivaUscitaTamburo\_change()

13 - Public Sub ControlloCadutaTamburoFiamma(tamburo As Integer)

13 - Public Function ValoreForchetta(valore As Double, min As Double, max As Double, SoloPositivo As Boolean) As Double

12 - Public Sub FiltroModulatore\_change()

12 - Public Sub AriaTamburoModulatore\_change(tamburo As Integer, apre As Boolean, chiude As Boolean)

12 - Public Sub AriaFreddaFiltroModulatore\_change()

9 - Public Sub TempIngressoTamburo\_change()

9 - Public Sub OlioCombInTemperatura\_change(tamburo As Integer)

9 - Public Sub BruciatorePosizioneAccensione\_change(tamburo As Integer)

9 - Public Sub AbilitaAvvCaldo()

8 - Public Sub TempScambComb\_change()

7 - Public Sub PressioneInsufficienteOlioCombustibile\_change(tamburo As Integer)

5 - Public Sub StartBruciatoreDaPLC\_change(tamburo As Integer)

4 - Public Sub PosizioneModulatoreAriaFredda\_change()

ParaTabBruc.bas

213 - Public Function ParaTabBruc\_ReadFile() As Boolean

88 - Public Sub ParaTabBruc\_Apply()

17 - Private Function String2Fuel(fuel As String) As FuelType

#### Oggetti utilizzati nella gestione essiccatore:

ControlloBruc.bas

Public Enum ModulatoreStatusEnum

Definisce gli stati possibili di un modulatore (up/down/none)

Public Type ModulatoreType

Il modulatore è una gestione logica riconducibile ad una valvola, il controllo avviene sulla apertura/chiusura (up/down), il dato di verifica è la posizione reale della valvola, queste grandezze sono espresse in valore percentuale.

Public Type TypeGestioneFumiTamburo  
Rappresenta la gestione del modulatore della valvola di parzializzazione dei fumi all’interno del tamburo, tale gestione è manuale vedi CmdUpDownBruc\_MouseDown() CmdUpDownBruc\_MouseUp().

Public Type TypeGestioneVelocitaTamburo

Rappresenta la gestione della modulazione della velocità del tamburo, tale gestione è manuale vedi CmdUpDownBruc\_MouseDown() CmdUpDownBruc\_MouseUp().