### Gestione Predosaggio

#### Descrizione

* Logica di funzionamento divisa tra PLC e applicativo
* CYB500 prevede funzioni di controllo, visualizzazione e impostazione del lavoro dei predosatori:
  + switch gestione automatica/manuale
  + calcolo di alcuni parametri (umidità, …)
  + controlli sulla catena di funzionamento degli apparati (start e stop dei predosatori e dei nastri)
  + gestione ricette da produrre (interazione con database)
  + minima interazione dosaggio-predosaggio; blocco i predosatori se ho finito di fare impasti.
  + Impostazione dei valori di portata dei predosatori

#### Parametri

Sul caricamento della Form e sulla applicazione di eventuali cambiamenti vengono gestiti i parametri presenti nel file xml (Parameters.xml)

{

e nei file .ini (Cisterne.ini). Le variabili da scambiare col PLC si trovano nel file OPCTags.xls.

}

#### Logica

##### Avviamento Predosatori

In vari punti del codice viene valutata la possibilità di far funzionare i predosatori in automatico o manuale. In generale, per lavorare in automatico, le condizioni sono:

* Motore di rotazione dell’essiccatore acceso
* Bruciatore acceso

##### Retroazione Dosatori-Predosatori

Quando i cicli di dosaggio da eseguire meno i cicli di dosaggio eseguiti raggiungono un valore impostato nei parametri o dall’operatore (CicliStopPred), blocco i predosatori. In pratica, se in dosaggio non ho bisogno di altro materiale per finire i miei cicli, non c’è bisogno che i predosatori alimentino la torre di dosaggio.

##### Gestione ricette da produrre

Le ricette che possono essere prodotte sono ricavate dal database, espresse in percentuali di peso) e possono essere selezionate dall’operatore. In base alla ricetta scelta i predosatori dovranno rilasciare una certa quantità di materiale.

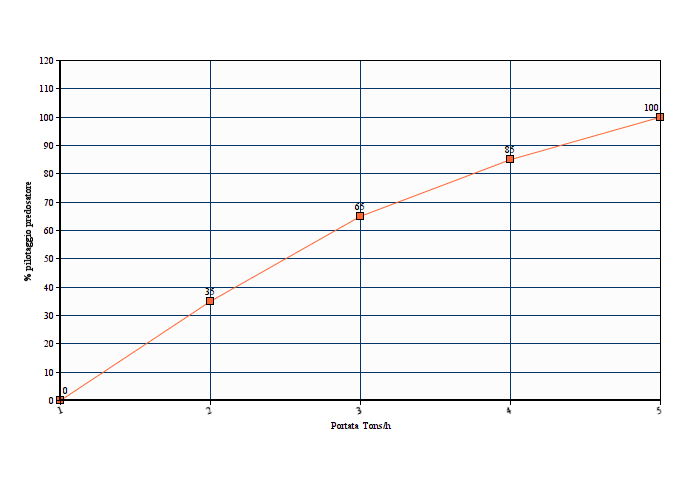
Scelta Ricetta

PLC

Calcolo Uscita Predosatori

Predosaggio

Il materiale in uscita dal predosatore non ha un andamento proporzionale con il settaggio impostato, ma segue un grafico, approssimato nel programma con una interpolazione lineare a 5 punti. In base alla portata richiesta dalla ricetta calcolo a che percentuale di pilotaggio far funzionare il predosatore.



##### Nastro del riciclato

Il deflettore del riciclato può essere impostato su 2 valori:

* Anello: il riciclato è caldo e finisce nel tamburo
* Elevatore: il riciclato è freddo e va direttamente in torre

Se abilitato il deflettore a Elevatore/Anello (AbilitaDeflAnElev su parameters.xml)

Nel caso di riciclato caldo, si calcola una retroazione PID (proporzionale-integrativo-derivativo) sul nastro. Prima calcolo il peso teorico del riciclato:

{TODO togliere l’equazione, fatta per provare}

La bilancia mi da il peso reale. A quel punto applico il PID e trovo il valore da applicare al nastro come retroazione.

### Dettaglio Funzioni

##### ParaTabPred.bas

Funzioni per lettura e caricamento parametri

##### GestionePredNastriC.bas

Principali funzioni:

* DatiSetPredosaggi()  
  Calcola l’umidità totale dei predosatori, sul comando di start dei predosatori e sul cambio stato degli stessi
* AvviamentoGestionePredosatori()  
  Gestisce l’avviamento dei predosatori. Se i predosatori sono in automatico e sono vergini accesi => se l’essiccatore non è acceso e i tamburi non sono caldi => mando allarme e passo a gestione manuale dei predosatori lanciando PassaInManualePredosatori().
* ControlloPredAutomatico(nastro As Byte) As Boolean  
  Controlla che i predosatori non partano se il nastro collettore è fermo e la gestione è semiautomatica
* PredosatoriVerginiAccesi() As Boolean  
  Verifica se almeno un predosatore di materiale vergine è acceso.
* PredosatoriRiciclatiAccesi() As Boolean  
  Verifica se almeno un predosatore del riciclato è acceso.
* GestioneStopPredosatori()  
  Gestisce lo stop dei predosatori. Viene lanciata quando i cicli di dosaggio da eseguire meno i cicli di dosaggio eseguiti raggiungono un valore impostato nei parametri o dall’operatore (CicliStopPred).
* ControllaRicettaPredVuota()  
  Controlla se la ricetta è vuota. Se è vuota, non trasferisce i set (percentuali di pilotaggio) CP240.OPCData.items(PLCTAG\_NM\_PRED\_RICFUT\_Inerte\_Set1).Value = 0
* AvvioPredAutomatico()  
  Fa partire i predosatori in automatico.
* PredosatoreSetCalcolato(ByRef Pred As PredosatoreType, portataTeorica As Double) As Integer  
  Calcola il valore da attribuire al Set dei predosatori n base alle tons/h (portata presa in cinque punti) tenendo presente l’andamento del grafico.   
  Si usa una interpolazione lineare in 5 punti. Grafico con portata sulle x e % di potenza al dosatore sulle Y. Data la portata teorica si imposta la % di lavoro, calcolandola con la linea tra due punti.
* ValoreUscitaAnalogicaPred(riciclato As Boolean, ValoreSet As Integer, ValoreSetReale As Integer) As Double  
  Usata da PredosatoreUscitaAnalogica per calcolare il valore della uscita analogica dei predosatori.  
  Premessa: ogni uscita o ingresso analogica del plc va da 0 a 27648, sia che si parli di 0-20 mA, 4-20 mA o 0-10V. Il valore massimo di scala (quello che consideriamo 1 in una scala da 0 a 1) si trova in Parameter.xml. Ad esempio il valore del voltaggio dei predosatori è dato dalla variabile Vpred.  
  <Vpred Type="2" TagOPC="TagDaSistemare" Default="8" MinValue="-10" MaxValue="+10" DescrID="504" TtipID="0" MeasUnitID="994" ReadAccess="1" WriteAccess="3" Printable="1" EditMode="4" />  
  Quindi nel nostro caso dei predosatori sappiamo che al massimo si accetta un voltaggio di 8V pari nel plc a 22118 (unità del predosatore).  
  Se ora moltiplico il valore di uscita analogico calcolato (che è una percentuale di potenza voglio andare) con questo valore ottengo il dato da spedire al PLC.
* PulsanteStopPred()  
  Mette in manuale i predosatori.
* RichiamoRicettaPredos()  
  Setta sui predosatori i dati della ricetta. Lavora sulla tabella di database Predosaggio, PredosaggioLog. Sono tabelle dove il lavoro dei predosatori è espresso con percentuali.
* AggiornaProssimaRicettaPredos()  
  Aggiorna a video la prossima ricetta.???
* PassaInManualePredosatori()  
  Fa passare la gestione dei predosatori in Manuale. Il tag PLC interessato è: PLCTAG\_NM\_PRED\_Auto\_Man. Se non è selezionata l’attesa di fine ricetta allora blocco i predosatori (PredosatoriArrestoImmediato) altrimenti la finisco.
* PredosatoriRicConBilancia() As Boolean  
  Dice se i predosatori del riciclato hanno bilancia
* *VisualizzaPortateNastri()*Visualizza o meno i controlli (in CP240) dedicati alla visualizzazione e memorizzazione della portata (effettiva o teorica) che transita sui nastri elevatori
* StartPredosatori\_change()  
  Aggiorna a video
* LeggeFileUmiditaPredosatoreRic(),LeggeFileUmiditaPredosatore ()  
  Leggono da un file specifico i dati dell’umidità dei predosatori. Sono i file Umidita-Predosatore.ini, Umidita-PredosatoreRic.ini.
* RinfrescaOrigineDatiPredosaggio(NomeRicPredSel As String, Optional AggiornaCP240 As Boolean), RinfrescaNomeRicPreDosaggio()  
  Aggiornamenti a video
* NastroRicRegolazionePonderale()  
  Abilita la regolazione ponderale del riciclato. Se ho le bilance su ogni riciclato posso calcolare il peso che teoricamente dovrebbe avere il nastro riciclato e col PID gestire la quantità da far uscire istantaneamente. Se ho un solo nastro controllo quello e basta.
* PesoNastroInerti(),PesoNastroRiciclato(),PesoNastroRiciclatoParDrum()  
  Chiamata su PlcInAnalogici, calcola il peso dei nastri di inerti, riciclato, riciclato al tamburo
* GestioneConsumi()  
  Invocata direttamente da VideataPrincipale; al max una chiamata al secondo. Aggiorna totalizzazione peso dei nastri elevatori; se la bilancia non è inclusa utilizza un valore teorico
* ResetTotalizzatoriNastri(nastro As Integer)  
  Manda al PLC il comando di reset del totalizzatore di un nastro e aggiorna a video il dato dei consumi.
* PredosatoriArrestoImmediato(riciclato As Boolean, nastro As Integer)  
  La funzione è invocata sul passaggio in manuale, su allarmi dei predosatori e su stop del bruciatore/tamburo. Arresta immediatamente una linea di predosaggio.

Moduli interessati alla gestione predosaggio

GestionePredNastriC.bas

<descrizione>

GestionePredosatori.bas

<descrizione>

ParaTabPred.bas

<descrizione>

Principali metodi utilizzati nella gestione essiccatore:

GestionePredNastriC.bas

Public Sub DatiSetPredosaggi()

Public Sub AvviamentoGestionePredosatori()

Public Function ControlloPredAutomatico(nastro As Byte) As Boolean

Public Function PredosatoriVerginiAccesi() As Boolean

Public Function PredosatoriRiciclatiAccesi() As Boolean

Public Sub GestioneStopPredosatori()

Public Function ControllaRicettaPredVuota()

Public Sub AvvioPredAutomatico()

Public Sub PreparazioneAvvPred()

Public Function PredosatoreSetCalcolato(ByRef Pred As PredosatoreType, portataTeorica As Double) As Integer

Public Function ValoreUscitaAnalogicaPred(riciclato As Boolean, ValoreSet As Integer, ValoreSetReale As Integer) As Double

Public Sub PulsanteStopPred()

Public Sub RichiamoRicettaPredos()

Public Sub AggiornaProssimaRicettaPredos()

Public Sub PassaInManualePredosatori()

Public Function PredosatoriRicConBilancia() As Boolean

Public Sub VisualizzaPortateNastri()

Public Sub StartPredosatori\_change()

Public Sub LeggeFileUmiditaPredosatore()

Public Sub LeggeFileUmiditaPredosatoreRic()

Public Sub RinfrescaOrigineDatiPredosaggio(NomeRicPredSel As String, Optional AggiornaCP240 As Boolean)

Public Sub RinfrescaNomeRicPreDosaggio()

Public Sub NastroRicRegolazionePonderale()

Public Sub PesoNastroInerti\_change()

Public Sub PesoNastroInerti()

Public Sub PesoNastroRiciclato\_change()

Public Sub PesoNastroRiciclatoParDrum\_change()

Public Sub PesoNastroRiciclato()

Public Sub PesoNastroRiciclatoParDrum()

Public Function SelezioneRicettaPredosaggioCambiata() As Boolean

Public Function PrimoPredosatoreDelNastro(nastro As NastriPredosatori) As Integer

Public Sub GestioneConsumi()

Public Sub ResetTotalizzatoriNastri(nastro As Integer)

GestionePredosatori.bas

Public Sub VisualizzaPredosatoriImpostati()

Public Sub VisualizzaPredosatoriRicImpostati()

Public Sub PredosatoreInizializza()

Public Sub PredosatoreUscita\_change(ByVal predosatore As Integer)

Public Sub SetPredosatoreUscita(ByVal predosatore As Integer, valore As Boolean)

Public Sub CalcolaNumeroPredosatoreSiwarex()

Public Sub PredosatoreRitorno\_change(ByVal predosatore As Integer)

Public Sub PredosatoreRicUscita\_change(ByVal predosatore As Integer)

Public Sub SetPredosatoreRicUscita(ByVal predosatore As Integer, valore As Boolean)

Public Sub PredosatoreRiciclatoRitorno\_change(ByVal predosatore As Integer)

Public Sub PredosatoreMinimoVuoto\_change(ByVal predosatore As Integer)

Public Sub PredosatoreRiciclatoMinimoVuoto\_change(ByVal predosatore As Integer)

Public Function PredosatoriCalcoloUmiditaTotaleTmp(Umidita() As Double, tutti As Boolean) As Double

Public Function PredosatoriCalcoloUmiditaTotale(tutti As Boolean) As Double

Public Function PredosatoriRiciclatoCalcoloUmiditaTotaleTmp(Umidita() As Double, Index As Integer) As Double

Public Function PredosatoriRiciclatoCalcoloUmiditaTotale(Index As Integer) As Double

Public Sub RitardoVibratorePredVuoto()

Public Sub RitardoSoffioPredVuoto()

Public Sub TermicaPredosatori\_change()

Private Function PredosatoreRitardato(ByRef Pred As PredosatoreType) As Boolean

Public Function PredosatoreOttieniSet(riciclato As Boolean, ByVal predosatore As Integer) As Integer

Public Sub PredosatoreCambiaSet(riciclato As Boolean,ByVal predosatore As Integer, valore As Integer,cambioManuale As Boolean)

Public Function PIDcontroller( \_

Public Sub PredosatoreInVolumetrico(riciclato As Boolean, predosatore As Integer, attivo As Boolean)

Public Sub PredosatoreVerificaSet(ByRef Pred As PredosatoreType, forza As Boolean)

Public Sub PredosatoriSet\_timer()

Public Sub PredosatoreManuale(riciclato As Boolean, predosatore As Integer, start As Boolean, forza As Boolean)

Public Sub PredosatoriInStartAutomatico(start As Boolean)

Public Sub PredosatoriInManuale()

Public Sub PredosatoriRicalcolaSet()

Public Function PredosatoriAccesi(riciclato As Boolean, nastro As Integer) As Boolean

Public Sub PredosatoriArrestoImmediato(riciclato As Boolean, nastro As Integer)

Public Sub VisualizzaRiduzioneProduzione()

Public Sub SetRiduzioneProduzione(nuovaRiduzione As Integer)

Public Function CalcolaSetNastri(Index As Integer) As Double

Public Function PredosatoreOttieniMaterialeLogId(ByRef Pred As PredosatoreType) As Long

Public Function PredosatoreOttieniNome(ByRef Pred As PredosatoreType) As String

Public Function PredosatoreOttieniPortata(ByRef Pred As PredosatoreType) As Integer

Public Sub GestioneStoricoPredosaggioAggiungi()

Public Sub GestioneStoricoPredosaggio()

Public Sub ChkResetColorMateriali()

Public Function ChkCoherenceMaterial(IdPredosaggio)

Public Sub SelectFeederRecipeByCS(IdRecipe As Long)

Public Function ControlloCondizioniStartPreDosaggio() As Boolean

Public Sub StartPreDosaggio()

ParaTabPred.bas

Public Function ParaTabPred\_ReadFile() As Boolean

Public Sub ParaTabPred\_Apply()

Public Sub CaricaDatiGrafPredosatore(ByRef Pred As PredosatoreType)

Public Sub PlcGriglieVibranti()

Public Sub LeggiUmiditaPredSQL()

Principali oggetti utilizzati nella gestione essiccatore:

GestionePredNastriC.bas

Public Type TolleranzaNastro

GestionePredosatori.bas

Public Type SetPredosatoreType

Public Type PidType

Public Type CurvaPredosatoreType

Public Type GraficoPredosatoreType

Public Type MotorePredosatoreType

Public Type PredosatoreType

Public Type FeederVibratorWorkingCycle\_TYPE

GestionePredNastriC.bas

Public Enum NastriPredosatori

GestionePredosatori.bas

Public Enum CambioSetPredosatoreType

Public Enum StatoPredosatoreType

Public Enum BilanciaNastro