

```

classdef ControllerOnOff < Controller
    properties %Controller State stored here.
        last_u
        Umax
        Umin
        errorband
    end %properties
    methods

        function obj=ControllerOnOff(errorband)
            arguments
                errorband=0;
            end
            obj.Umax=1.5;
            obj.Umin=0;
            obj.last_u=obj.Umin;
            obj.errorband=errorband;
        end

        function Track(obj,sensor>manual_u,T)
            %do nothing; code is needed for "bumpless transfer".
        end

        function u=ComputeU(obj,setpoint,sensor,T)
            error=setpoint-sensor;
            if obj.errorband==0
                u=obj.NaiveOnOff(error);
            else
                u=obj.OnOffWithBand(error);
            end
        end

        function u=NaiveOnOff(obj,error)
            u=obj.Umax*(error>0)+obj.Umin*(error<=0);
            %This is equivalent to an "if" statement: if error>0 u=Umax; else u=Umin; end
        end

        function u=OnOffWithBand(obj,error)
            %The one to use in most cases in pressure/level/temperature control applications
            band=obj.errorband;
            if error>band
                u=obj.Umax;
            elseif error<=-band
                u=obj.Umin;
            else
                u=obj.last_u;
            end
            obj.last_u=u;
        end

    end %methods
end %classdef

```

Video (YouTube, Spanish):

<http://personales.upv.es/asala/YT/V/tank10n0ff.html>