

Homodyne receiver

March 26, 2017

This block of code simulates the reception and demodulation of an optical signal (which is the input signal of the system) outputting a binary signal. A simplified schematic representation of this block is shown in figure 1.

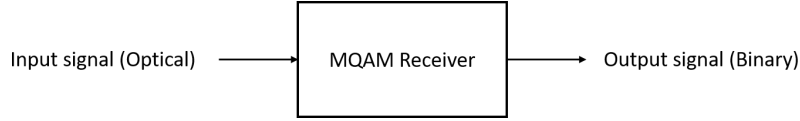


Figure 1: Basic configuration of the MQAM receiver

Functional description

This block accepts one optical input signal and outputs one binary signal that corresponds to the M-QAM demodulation of the input signal. It is a complex block (as it can be seen from figure 2) of code made up of several simpler blocks whose description can be found in the *lib* repository.

It can also be seen from figure 2 that there's an extra internal (generated inside the homodyne receiver block) input signal generated by the *Clock*. This block is used to provide the sampling frequency to the *Sampler*.

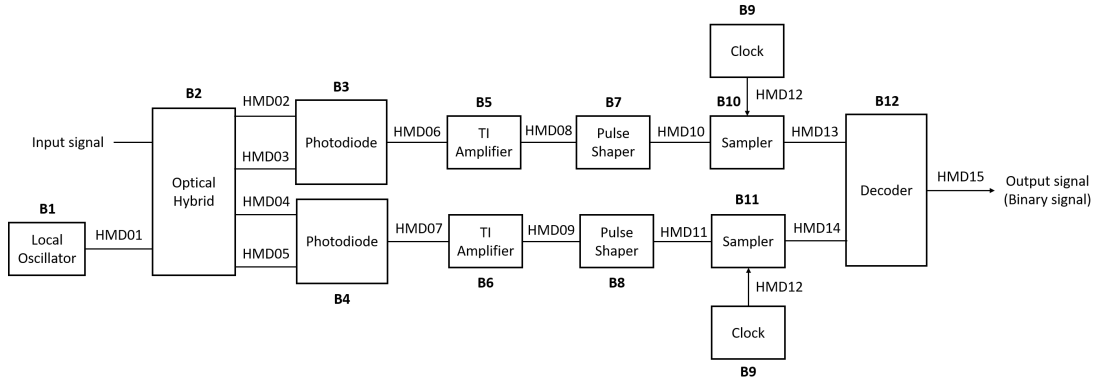


Figure 2: Schematic representation of the block homodyne receiver.

Input parameters

- `samplingPeriod{0.0}`
- `localOscillatorOpticalPower{ 1e-3 };`
- `localOscillatorPhase{ 0 };`
- `localOscillatorWavelength{ 1550e-9 };`
- `outputOpticalFrequency SPEED_OF_LIGHT / localOscillatorWavelength ;`

Methods

HomodyneReceiver(vector<Signal*> &inputSignal, vector<Signal*> &outputSignal) (**constructor**)

void setIqAmplitudes(vector<t_iqValues> iqAmplitudesValues)

vector<t_iqValues> const getIqAmplitudes(void)

void setLocalOscillatorSamplingPeriod(double sPeriod)

void setLocalOscillatorOpticalPower(double opticalPower)

void setLocalOscillatorOpticalPower_dBm(double opticalPower_dBm)

void setLocalOscillatorPhase(double lOscillatorPhase)

void setLocalOscillatorOpticalWavelength(double lOscillatorWavelength)

void setSamplingPeriod(double sPeriod)

void setResponsivity(t_real Responsivity)

void setAmplification(t_real Amplification)

void setNoiseAmplitude(t_real NoiseAmplitude)

void setImpulseResponseTimeLength(int impResponseTimeLength)

void setFilterType(PulseShaperFilter fType)

void setRollOffFactor(double rOffFactor)

void setClockPeriod(double per)

void setSamplesToSkip(int sToSkip)

Input Signals

Number: 1

Type: Optical signal

Output Signals

Number: 1

Type: Binary signal

Example

Sugestions for future improvement