6.1) Introduction

Interferometers are optical components that split lightwaves into two waves by means of a beamsplitter, delay these waves by unequal distances, redirect and recombine them by means of another (or the same) beamsplitter, and detect the intensity of their superposition. Mach-Zehnder interferometers (MZIs) are one of three important variants of interferometers, with the others being the Michelson interferometers and Sagnac interferometers (note that the above three configurations are not the only interferometers available). MZIs are widely deployed as fundamental building blocks in planar lightwave circuits (PLCs).

Figure 6.1 shows the two MZI variants in PLC form. The asymmetrical MZI achieves a delay as described above by varying the path length between the two waveguides connecting the couplers. The couplers act as the beamsplitters discussed earlier. The asymmetrical MZI can be used for multiplexing/demultiplexing signals in the GHz range for channel spacing [1]. The symmetrical MZI employs a heater (an electrode can also be used) in order to achieve the delay between the two arms. This MZI is suitable for the manufacturing of tuneable couplers [2].

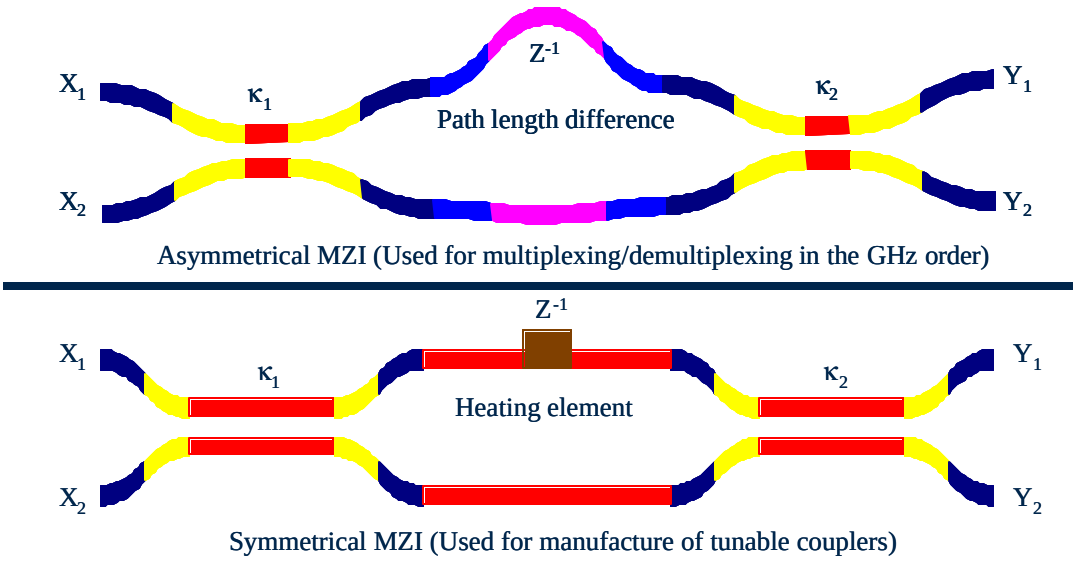

Figure 6.1: Possible MZI configurations

6.2) Z-transform description of MZIs

MZIs can be described in terms of the z transform [2]. This transformation method is especially handy when the MZIs are concatenated to form lattice filters, because it allows for direct implementation of recursion formulas [3] (discussed in section 6.5.1). In figure 6.2, a block diagram description of the MZI as a z-transformed filter is presented.

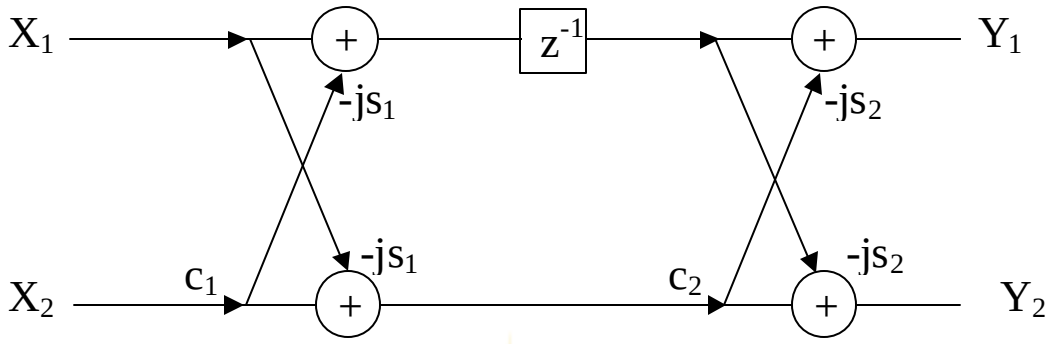

Figure 6.2: Block diagram description of MZI

As can be seen, the one arm has a transfer function of z^{-1} relative to the other arm. In the case of the asymmetrical MZI, it means that the longer arm has a transfer function of z^{-1} relative to the shorter one. The z^{-1} term represents a unit delay, which in the case of the asymmetrical MZI, is given by

$$L_U = \Delta L = L_1 - L_2 \tag{6.1}$$

where L_1 and L_2 represent the lengths of the waveguide paths connecting the couplers.

A 2×2 delay matrix is written as

$$\Phi_{delay} = g e^{-jbL_2} \begin{bmatrix} z^{-1} & 0 \\ 0 & 1 \end{bmatrix} \tag{6.2}$$

where $-20log_{10}g = aL_2$ is the dB loss for propagation through the common path length. The term α represents the fundamental absorption loss described by $I(z) =$

$I_0(z)e^{-\alpha z}$ where I is the optical intensity and z the propagation distance. In general, $L_U << L_2$, so for simplification, the loss in the differential path length is neglected. In addition, the linear phase resulting from the common path length is neglected because we are only interested in the phase *delay*. The overall transfer matrix is the product of the scattering and delay matrices as derived from figure 6.2, as in equation 6.3

$$\begin{aligned}\Phi_{MZI} &= \Phi_{coup}(k_2)\Phi_{delay}\Phi_{coup}(k_1) \\ &= g\begin{bmatrix} c_2 & -js_2 \\ -js_2 & c_2 \end{bmatrix}\begin{bmatrix} z^{-1} & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} c_1 & -js_1 \\ -js_1 & c_1 \end{bmatrix} \\ &= g\begin{bmatrix} \left(-s_1s_2 + c_1c_2z^{-1}\right) & -j\left(c_1s_2 + s_1c_2z^{-1}\right) \\ -j\left(s_1c_2 + c_1s_2z^{-1}\right) & \left(c_1c_2 - s_1s_2z^{-1}\right) \end{bmatrix}\end{aligned} \tag{6.3}$$

where $s_n = (k_n)^{0.5}$ denotes the cross-coefficient and $c_n = (1 - k_n)^{0.5}$ denotes the through-coefficient in the scattering matrix. As can be seen, the MZI has one zero for each transfer function and is the simplest moving average optical filter (MAOF) [2].

The transfer functions for the MZI can now be defined as

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} H_{11}(z) & H_{12}(z) \\ H_{21}(z) & H_{22}(z) \end{bmatrix}\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} A(z) & B^R(z) \\ B(z) & A^R(z) \end{bmatrix}\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \tag{6.4}$$

Note that the coefficients of $H_{22}(z)$ are the reverse of those of $H_{11}(z)$. Practically, this means that the zero has been reflected around the unit circle. So, for example, if

$$H_1(z) = \Gamma(1 - az^{-1}) = h_0 + h_1z^{-1} \tag{6.5}$$

where G represents the gain and a is a polynomial, and we let $a \rightarrow 1/a^*$ then

$$H_1^R(z) = \Gamma(-z_1^* + z^{-1}) = h_1^* + h_0^*z^{-1} \tag{6.6}$$

In general, then, we find that

$$H_N(z) = h_0 + h_1z^{-1} + \cdots + h_Nz^{-N} \Rightarrow H_N^R(z) = h_N^* + h_{N-1}^*z^{-1} + \cdots + h_0^*z^{-N} \tag{6.7}$$

Returning to the MZI transfer function, we can now make some general observations by inspecting the transfer matrices. Firstly, we state that the conservation of power for a lossless filter requires that $|H_{11}(z)|^2 + |H_{21}(z)|^2 = |H_{12}(z)|^2 + |H_{22}(z)|^2 = 1$. Hence, we deduce that a zero on the unit circle for either the through- or cross-port response implies 100% transmission for the other response.

Secondly, if the coupling ratios are identical ($k_1 = k_2$), the zero of the cross-port response occurs on the unit circle. On the other hand, if $k_1 = 1 - k_2$, the zero of the through-port occurs on the unit circle. A special case arises when $k_1 = k_2 = 0.5$ (it now satisfies both of the abovementioned conditions), where the zeros of both the cross- and through-ports lie on the unit circle.

Figure 6.3 shows the result of a software simulation with $k_2 = 0.7$ and $k_1 = 1 - k_2 = 0.3$. Figure 6.4 shows the result of a simulation with $k_1 = k_2 = 0.3$ and, finally, figure 6.5 showing the result of the special case when $k_1 = k_2 = 0.5$.

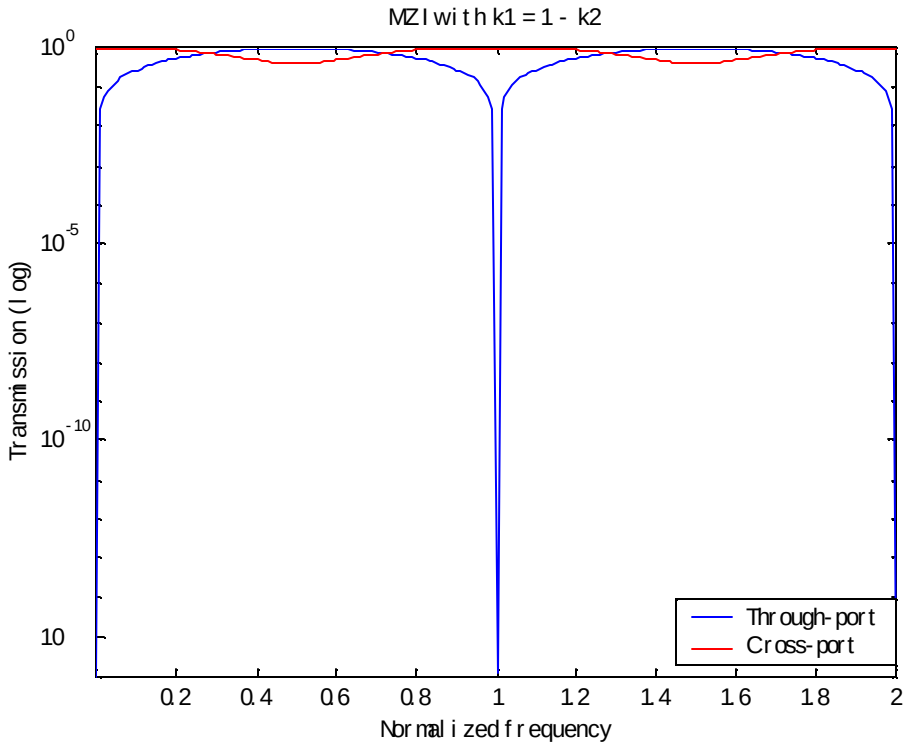

Figure 6.3: Idealised simulation of MZI with $k_1 = 1 - k_2$

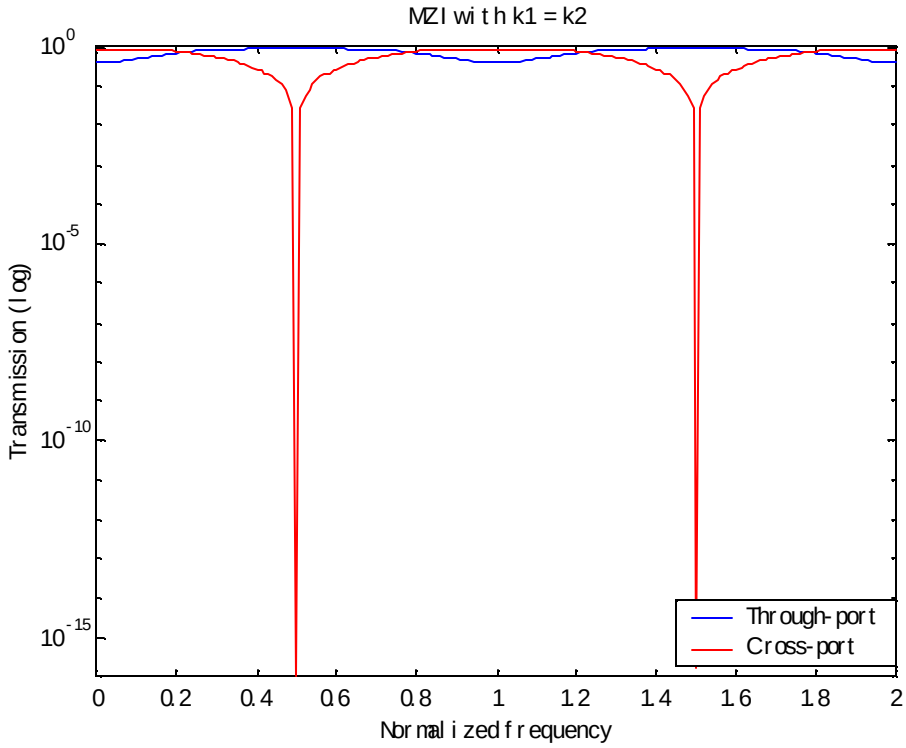

Figure 6.4: Idealised simulation of MZI with $k_1 = k_2$

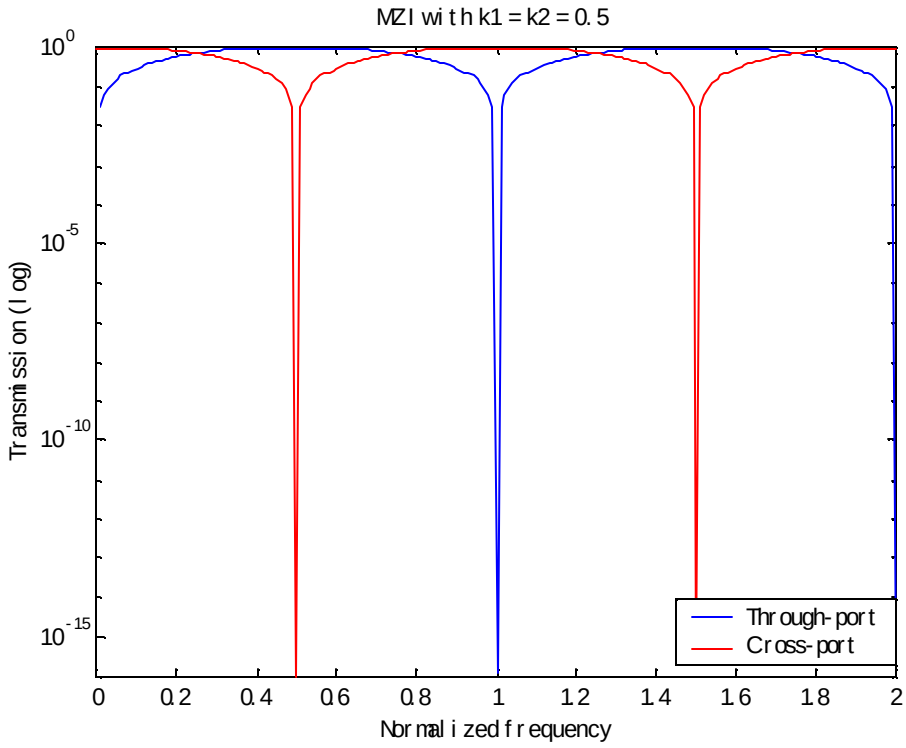

Figure 6.5: Idealised simulation of MZI with $k_1 = k_2 = 0.5$

From the above three figures it can be seen that by varying the coupling ratios, different positions can be obtained for the zero positions. This allows for the realisation of arbitrary filters as is discussed in section 6.5.

6.3) Loss and manufacturing variations

Losses (we recall that γ represented loss) for the SiON waveguides were shown to be 0.2dB/cm in section 4.3, so the differential path length must be many centimetres long to introduce a significant loss. Figure 6.6 shows a simulation for an MZI with 1dB loss present in the delay length. This loss was incorporated by multiplying a loss term with $\dot{z}^{1}$. Furthermore, the loss of the common path length as was discussed in the previous section was neglected for this simulation. It can be seen that the transmission nulls are shallower than those for the lossless case because the zeros are no longer on the unit circle.

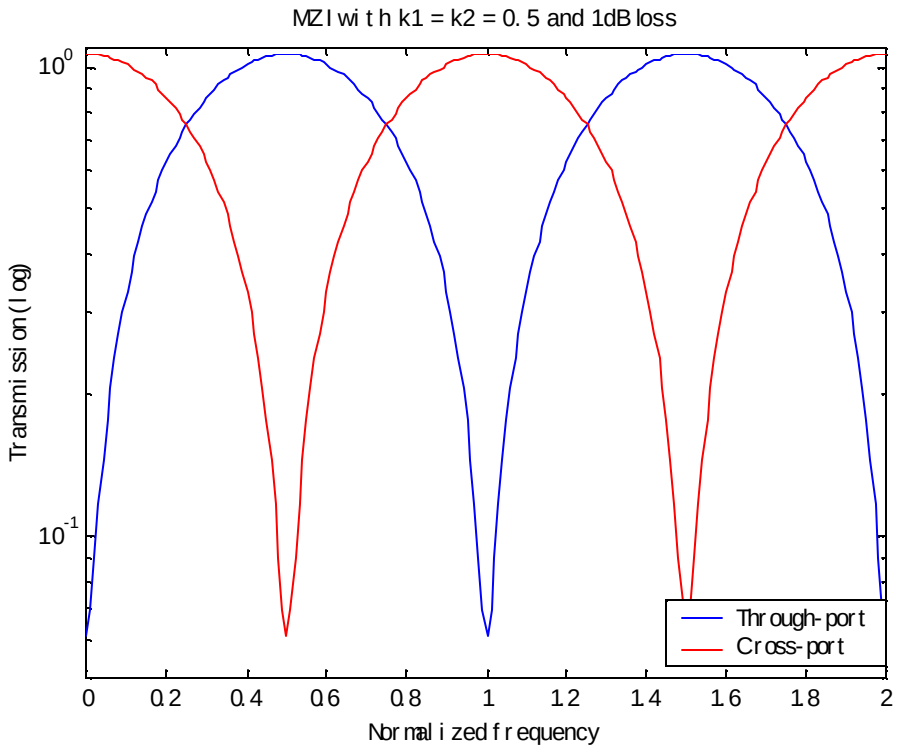

Figure 6.6: Idealised simulation of MZI with 1dB loss

When manufacturing an MZI, the mask used for patterning the circuit on the core material (the material with the relatively higher refractive index) has a finite resolution. The smaller the mask grid size, the smaller the discrepancies between the designed and physical MZIs in terms of the path length differences. These discrepancies can at best be minimized but not completely eradicated. Furthermore, random effective index variations can occur due to small deposition variations in terms of deposition thickness and the deposited refractive index. These variations in length and effective index cause phase errors in the device.

The abovementioned phase errors can be included in the transfer matrix in equation 6.3 by multiplying an additional term e^{-jf} (where f represents the additional phase change) into all the coefficients where z^{-1} occurs. In figure 6.7, a simulation is presented of an MZI with a phase fluctuation of $p/2$. It can be seen that the transmission nulls have shifted in frequency (to the right when compared with figure 6.5).

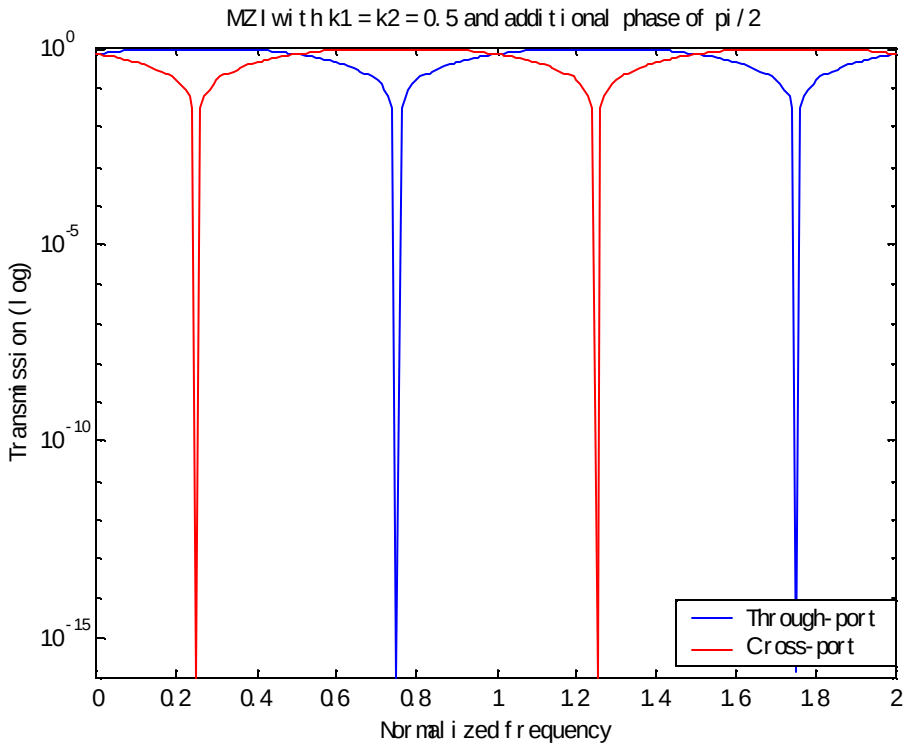

Figure 6.7: Idealised simulation of MZI with additional phase p/2

A practical way of compensating for these phase errors is to use the thermo-optic effect after manufacturing the MZIs. Thermo-optic heaters can be sputtered onto one of the waveguides, and can then be used to tune the phase back to the designed position. Another option is to compensate permanently for index variations through UV exposure of one of the waveguides.

6.4) Tuneable couplers by means of symmetrical MZIs

As stated earlier, symmetrical MZIs can be used to manufacture tuneable couplers [1,2,5]. The derivation of the required conditions starts with the assumption that the two couplers have the same coupling ratio, i.e. $k_1 = k_2 = k$. Substituting these values back into equation 6.3, and incorporating an additional term e^{-jf}, as discussed in the previous section, yields

$$\Phi_{MZI} = g\begin{bmatrix} \left(-k + (1-k)e^{-jf}z^{-1}\right) & -j\left(\sqrt{(1-k)k} + \sqrt{(1-k)k}\,e^{-jf}z^{-1}\right) \\ -j\left(\sqrt{(1-k)k} + \sqrt{(1-k)k}\,e^{-jf}z^{-1}\right) & \left((1-k) - e^{-jf}kz^{-1}\right) \end{bmatrix} \quad (6.8)$$

Setting $X_1 = 1$ and $X_2 = 0$ and assuming $g = 0$ (for the lossless case) yields

$$\begin{aligned} Y_1 &= (1-k)e^{-jf}z^{-1} - k \\ Y_2 &= -j\sqrt{(1-k)k}\left(e^{-jf}z^{-1} + 1\right) \end{aligned} \quad (6.9)$$

The square magnitude responses are given by

$$\begin{aligned} |Y_1|^2 &= (1-k)^2 + k^2 - 2k(1-k)\cos(f) \\ |Y_2|^2 &= 4k(1-k)\cos^2\left(\frac{f}{2}\right) \end{aligned} \quad (6.10)$$

The effective coupling ratio for the MZI is given by

$$k_{MZI} = \frac{|Y_2|^2}{|Y_1|^2 + |Y_2|^2} = |Y_2|^2 \quad (6.11)$$

It can be shown that $|Y_2|^2$ has a maximum value for $k = 0.5$. Furthermore, by letting $f = p$ (the maximum effective phase difference between the two arms) $|Y_2|^2 = 0$, independent of k. It is thus shown that introducing a phase change can control the power coupling [2]. Symmetric MZIs are used with phase shifters such as heaters to provide this phase control.

As an example, if we require a power split of exactly 50%, we set $|Y_2|^2 = 0.5$ and obtain the following condition:

$$k(1-k) \geq \frac{1}{8} \tag{6.12}$$

giving a boundary of $0.1464 \leq k \leq 0.8536$. In other words, to realize a coupler with an exact power split of 50%, the coupling coefficients of the two couplers making up the MZI must lie in the specified boundary.

It is important to note that the phase difference in the case of a tuneable coupler is dependent on the values of k, and is not a constant $p/2$. The relative phase is then described by

$$f_{rel} = \arg\left[\frac{Y_2}{Y_1}\right] \tag{6.13}$$

6.5) Lattice filters by means of asymmetrical MZIs

As was stated earlier, asymmetrical MZIs can be used as filters where channel separation is in the order of GHz [2,5]. Multistage lattice filters can be realized by concatenating MZIs as shown in figure 6.8. A major advantage of this architecture is that a very low loss passband can be achieved [3]. The unit delays result in the transmission being defined by a Fourier series. Lattice filters can then also be referred to as Fourier filters [4]. Lattice filters with specific transfer functions can be realized by designing MZIs with specific coupling ratios and unit path length differences. A very reliable method of obtaining these values is by using recursion formulas [2,3].

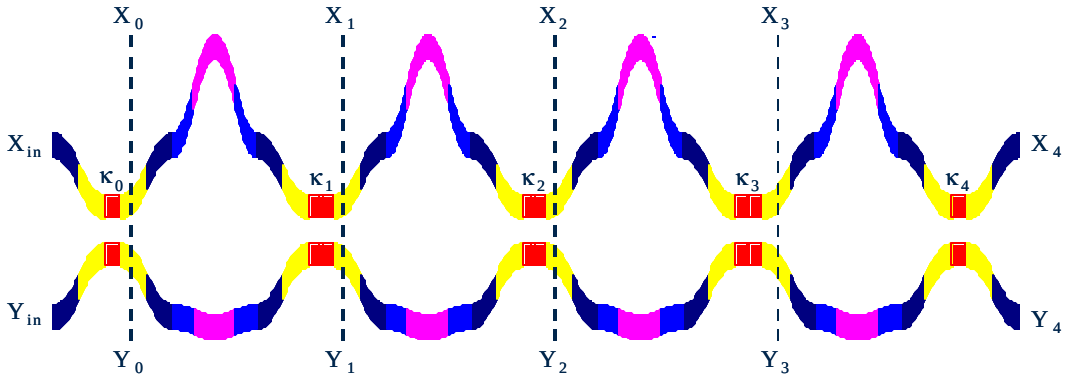

Figure 6.8: Layout of MZI lattice filter

6.5.1) Recursion formula and synthesis algorithm

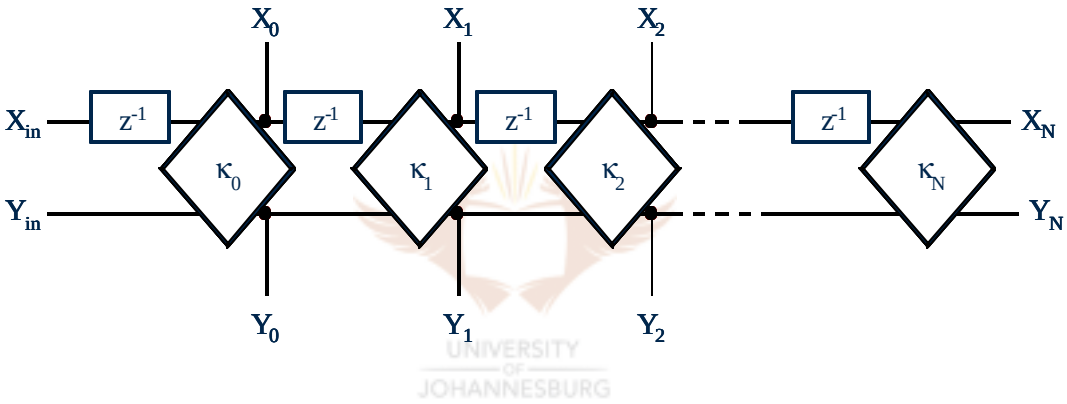

Figure 6.9: Block diagram for MZI lattice filter

In figure 6.9, a block diagram description of the lattice filter is presented as realised with an MZI structure. It is important to note that the input X_{in} has an added delay that is not realised in the physical manufactured filter. It does however simplify the form of the transfer matrix, and is removed when calculating the frequency response. From equations 6.3 and 6.4, we can derive the transfer matrix for the nth stage to be

$$\begin{bmatrix} X_n(z) \\ Y_n(z) \end{bmatrix} = \Phi_n(z) \begin{bmatrix} X_{n-1}(z) \\ Y_{n-1}(z) \end{bmatrix} \tag{6.14}$$

with

$$\Phi_n(z) = \begin{bmatrix} c_n e^{-jf_n} z^{-1} & -js_n \\ -js_n e^{-jf_n} z^{-1} & c_n \end{bmatrix}$$

For an Nth-order filter, the delay matrices are multiplied as follows:

$$\begin{bmatrix} X_n(z) \\ Y_n(z) \end{bmatrix} = \Phi_N \cdots \Phi_1 \Phi_0 \begin{bmatrix} X_{in}(z) \\ Y_{in}(z) \end{bmatrix} = \begin{bmatrix} z^{-1}A_N(z) & -jB_N^R(z) \\ -jz^{-1}B_N(z) & A_N^R(z) \end{bmatrix} \begin{bmatrix} X_{in}(z) \\ Y_{in}(z) \end{bmatrix} \tag{6.15}$$

The reverse polynomials are defined as

$$A_N^R(z) = z^{-N} e^{-jf_{tot}^N} A_N^*(z^{*-1}) \tag{6.16}$$

$$B_N^R(z) = z^{-N} e^{-jf_{tot}^N} B_N^*(z^{*-1}) \tag{6.17}$$

where $f_{tot}^N = \sum_{n=1}^{N} f_n$. Loss is included by substituting gz^{-1} for z^{-1}, implying the roots are reflected about a circle of radius g and not 1. It can be shown that the Nth-order polynomials are defined in terms of the lower order polynomials as [2]

$$\begin{bmatrix} z^{-1}A_N(z) & -jB_N^R(z) \\ -jz^{-1}B_N(z) & A_N^R(z) \end{bmatrix} = \begin{bmatrix} z^{-1}e^{-jfN}c_N & -js_N \\ -jz^{-1}e^{-jfN}s_N & c_N \end{bmatrix} \begin{bmatrix} z^{-1}A_{N-1}(z) & -jB_{N-1}^R(\\ -jz^{-1}B_{N-1}(z) & A_{N-1}^R(z \end{bmatrix} \tag{6.18}$$

For convenience the recursion formulas derived from equation 18 are now presented as

$$A_N(z) = c_N e^{-jfN} z^{-1} A_{N-1}(z) - s_N B_{N-1}(z) \tag{6.19}$$

$$B_N(z) = s_N e^{-jfN} z^{-1} A_{N-1}(z) + c_N B_{N-1}(z) \tag{6.20}$$

By expanding the above results, the following expressions are derived to describe the first and last coefficients of each polynomial

$$a_{0,N} = \begin{cases} -s_1 s_0 & for\ N = 1 \\ -s_N s_0 \prod_{n=1}^{N-1} c_n & for\ N > 1 \end{cases} \tag{6.21}$$

$$a_{N,N} = e^{-jf_{tot}^{N}} \prod_{n=1}^{N} c_n \tag{6.22}$$

$$b_{0,N} = s_0 \prod_{n=1}^{N} c_n \tag{6.23}$$

$$b_{N,N} = e^{-jf_{tot}^{N}} s_N \prod_{n=0}^{N-1} c_n \tag{6.24}$$

Using equations 21 – 24, we obtain

$$-\frac{a_{0,N}}{b_{0,N}} = \frac{b_{N,N}}{a_{N,N}} = \frac{s_N}{c_N} = \tan q_n \tag{6.25}$$

The coupling ratio can be derived [2] from equation 25 as

$$k_N = \frac{\left|\frac{b_{N,N}}{a_{N,N}}\right|^2}{1+\left|\frac{b_{N,N}}{a_{N,N}}\right|^2} = \frac{\left|\frac{a_{0,N}}{b_{0,N}}\right|^2}{1+\left|\frac{a_{0,N}}{b_{0,N}}\right|^2} \tag{6.26}$$

By multiplying equation 18 with $F_n^{-1}(z)$, the step-down recursions, which are also the fundamental equations for the algorithm, can be defined as

$$A_{N-1}(z) = ze^{jfN} \overset{dummy}{A} \tag{6.27}$$
$$with \quad \overset{dummy}{A} = \{c_N A_N(z) + s_N B_N(z)\}$$

$$B_{N-1}(z) = -s_N A_N(z) + c_N B_N(z) \tag{6.28}$$

It is useful to recap at his stage. We have derived the fundamental equations for the recursion formula in equations 6.27 and 6.28. To determine the s and c parameters, we are required to determine the coupling coefficient κ. This was done in equation 6.26. We note now that we still need to calculate the term f_N in equation 6.27. This is done by utilising equation 6.24 as follows:

$$f_{tot}^{N-1} = -\arg\left(b_{N-1,N-1}\right) \tag{6.29}$$

Furthermore, according to equation 6.22, the argument of the last coefficient must satisfy

$$\arg\left(a_{N-1,N-1}^{dummy}\right) + f_N = -\arg\left(a_{N-1,N-1}\right) = f_{tot}^{N-1} \tag{6.30}$$

So, substituting equation 6.29 into equation 6.30 yields

$$f_N = -\arg\left(a_{N-1,N-1}^{dummy}\right) - \arg\left(b_{N-1,N-1}\right) \tag{6.31}$$

Finally, because of the technical constraint that only one polynomial is normally specified by the design, e.g. $A_N(z)$, we need to find a valid $B_N(z)$ to use in the recursion formula. The expression derived in [2] is

$$B_N(z)B_N^R(z) = -A_N(z)A_N^R(z) + z^{-N}e^{-jf_{tot}^N} \tag{6.32}$$

The roots of equation 6.32 occur in reciprocal pairs around the unit circle (or around the γ circle if loss is included), one root from each pair belonging to $B_N(z)$ and the other to $B_N^R(z)$. Through the process of spectral factorisation, we may choose to assign the minimum phase roots to $B_N(z)$ and the maximum phase roots to $B_N^R(z)$. The roots of $B_N(z)$ only specify it to within a multiplicative constant α. This scaling factor is defined as follows

$$a = \sqrt{\frac{-a_{0,N}a_{N,N}}{b'_{0,N}b'_{N,N}}} \tag{6.33}$$

where the $b'_{x,x}$ terms represent the coefficients for the polynomial $B'_N(z)$ of the equation $B_N(z) = a\ B'_N(z)$.

The steps in designing an optical lattice filter from the expression of an Nth-order polynomial are summarised in figure 6.10 [2].

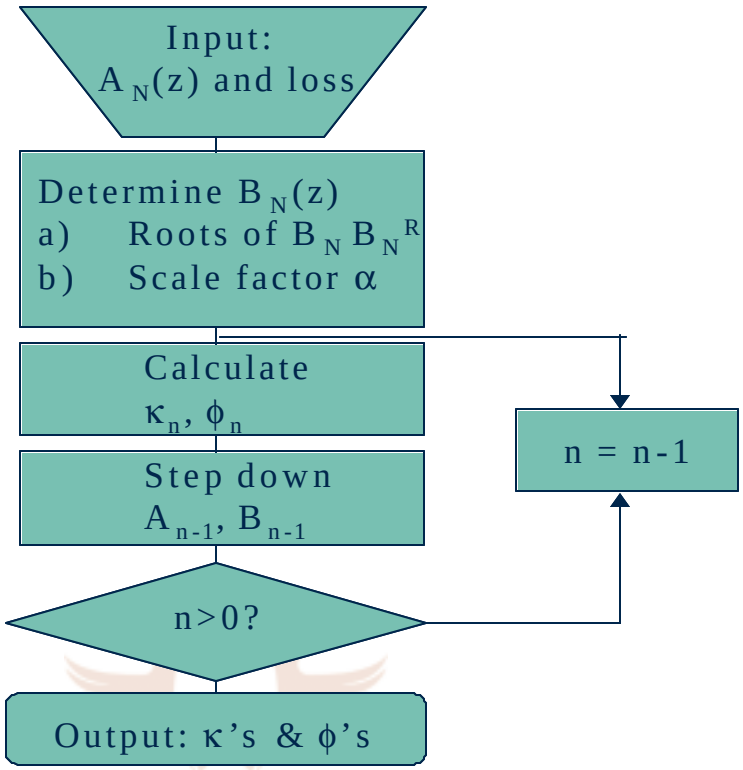

Figure 6.10: Flow diagram for recursion formula implementation

As a final note, it is important to stress that $B_N(z)$ may also be supplied as the required function, and although not derived here, the recursion formula (with some modifications) works just as well in this case. MATLAB programs implementing the recursion relations for both $A_N(z)$ and $B_N(z)$ inputs can be found in the appendix.

6.5.2) MZI design for lattice filters

In this section we explore the design criteria for the MZI itself to be used in a lattice filter. We start by restating that an MZI delays one wave relative to another in order to achieve some desired superposition. This delay represents a phase difference between the two waves. The phase for each path is the product of the distance travelled L and the propagation constant b, as is shown in equation 6.34.

$$f = Lb = L\frac{2pn_e}{l} \tag{6.34}$$

where n_e is the effective index of the waveguide, and l is the wavelength of the signal. In an MZI, each delay must be an integer multiple of a unit delay length L_U. The centre wavelength is defined so that L_U is equal to an integer number of wavelengths [2]

$$m l_c = n_e L_U \quad \text{with } m = 0,1,2\ldots \tag{6.35}$$

Furthermore, it can be shown that for a delay line with dispersion the free spectral range (FSR) is defined as [2]

$$\Delta l_{FSR} = \frac{l^2}{n_g L_U} \tag{6.36}$$

where

$$n_g = n_e(l_{center}) - l_{center}\left.\frac{dn_e}{dl}\right|_{l_{center}} \tag{6.37}$$

When designing an MZI to be utilized in a lattice filter, equation 6.36 is used to determine the required path length difference (L_U) for a required FSR. This L_U is then substituted back into equation 6.35, and tuned slightly in order to satisfy the criteria that m must be an integer. The abovementioned procedure is implemented in the MZI MATLAB programs found in the appendix.

The physical layout of the MZI can be cumbersome. It must be optimised so as to ensure that the length doesn't become excessively long. A MATLAB program that uses simple geometrical relations was written to automate the calculation of the required arc lengths and can be found in the appendix, as well as a schematic of a three stage MZI lattice filter.

6.6) Simulation: Tuneable MZI coupler

Simulations were performed in the BeamPROP environment to investigate the working of a tuneable coupler employing a symmetrical MZI and a heater on one of the arms. BPM simulations for an MMI tuneable coupler with a push/pull configuration were also conducted.

As was noted in section 4.3, the waveguide system under investigation has a thermo-optic coefficient of $dN_{eff}/dT = 1.2 \times 10^{-5}\ °C^{-1}$, referred to from now on as TC. Furthermore, the thermal conductivity of SiON is in the order of 1.6W/m.K. In BeamPROP, the thermal conductivity has to be described in terms of µm. Thus, the thermal conductivity, referred to as Kappa from now on, is equal to $1.6e^{-6}$W/µm.K.

TC and Kappa are introduced into BeamPROP via the Electrode/Heater table of the program. In this table, the various layers comprising the waveguide structure are defined in terms of height, Kappa and TC. For this simulation, 4 layers were defined in order to simulate the under-cladding, slab region, core region and over-cladding. The Kappas and TCs for all these layers were chosen to be equal, which is a valid assumption for a first-order simulation process in view of the fact that these parameters will vary slightly between the different layers. Once a prototype has been manufactured, measurements can be taken and optimisation can commence.

6.6.1) Symmetrical MZI with bi-directional couplers

Coupler 2 is used in this design to ensure that the evanescent fields in the straight arms of the MZI do not interfere with each other. The coupling ratio of the two couplers is chosen to be 0.5, implying a coupling length of 256.2µm. The position of the heater on the y-axis is defined so as to simulate a sputtered heater on the over-cladding as discussed in section 3.4, implying a y-axis height of 6.94µm above y = 0, where y = 0 is as defined as in figure 4.1. The physical layout is presented in figure 6.11.

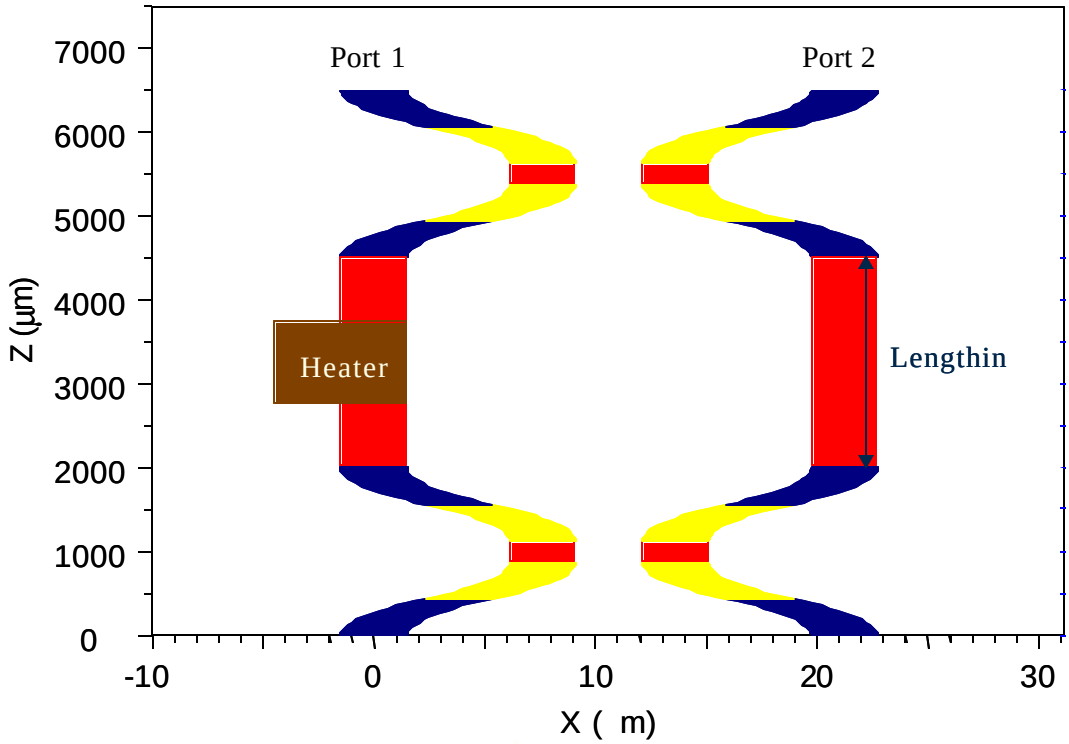

Figure 6.11: Layout of symmetrical MZI

In the first simulation, the temperature difference between the heater and the waveguide was incremented from 0K to 35K. In figure 6.12, the simulation results are presented. It is clear how the coupling ratio changes with applied temperature. If we required a coupling ratio of say 0.5, we see that an applied temperature of ±29K is required. Here the lengths of the arms were 2500µm, and the heater length was 1000µm. In order to optimise the design, two more simulations are presented next.

In the second simulation, the temperature difference was kept constant at 20K, and the ratio of the arm lengths to the heater length varied. The arm lengths were kept constant at 2500µm. This simulation is necessary to optimise the lengths of these two parameters. As can be seen in figure 6.13, a ratio of between 1 and 2 causes coupling ratio changes of 0.047 K^{-1} and 0.02 K^{-1} respectively.

In the third simulation, the temperature difference was kept constant at 20K, and the arm lengths varied while keeping the ratio of the arm lengths and the heater at a constant value of 2. As can be seen in figure 6.14, as the length is increased, the coupling ratio changes far more rapidly. This curve will typically be considered in

conjunction with the specifications for the control of the heating element. A conservative value for the arm length would be 2720μm, which implies a 0.025 coupling ratio change per K in temperature change.

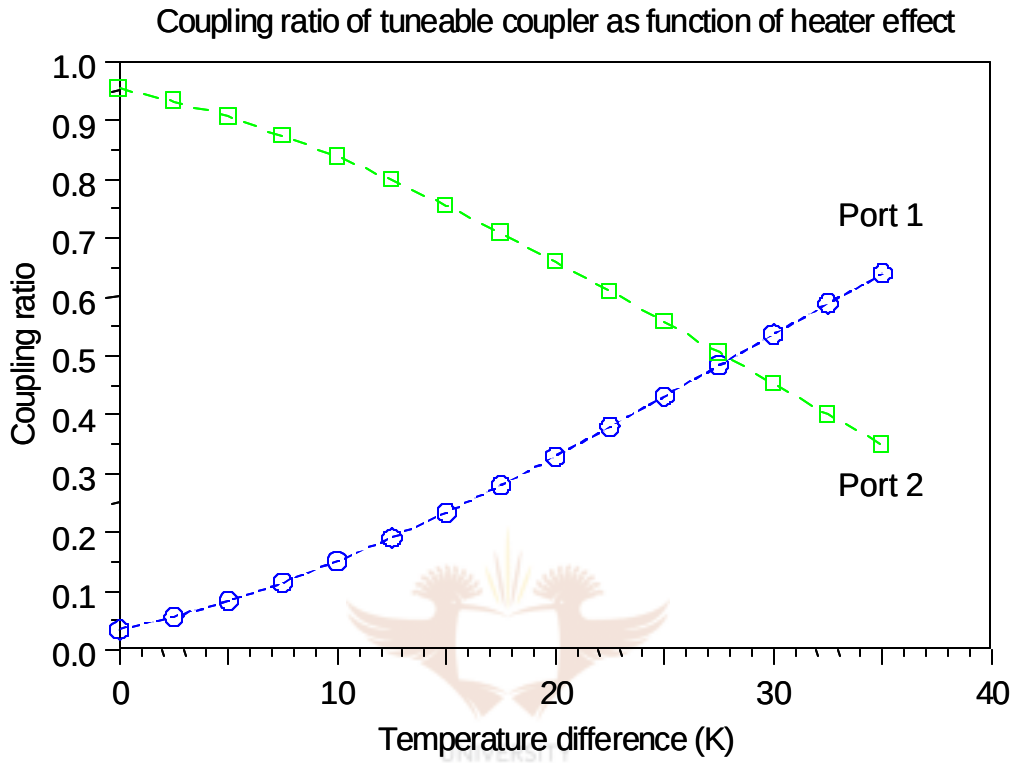

Figure 6.12: Typical output of symmetrical MZI with applied temperature

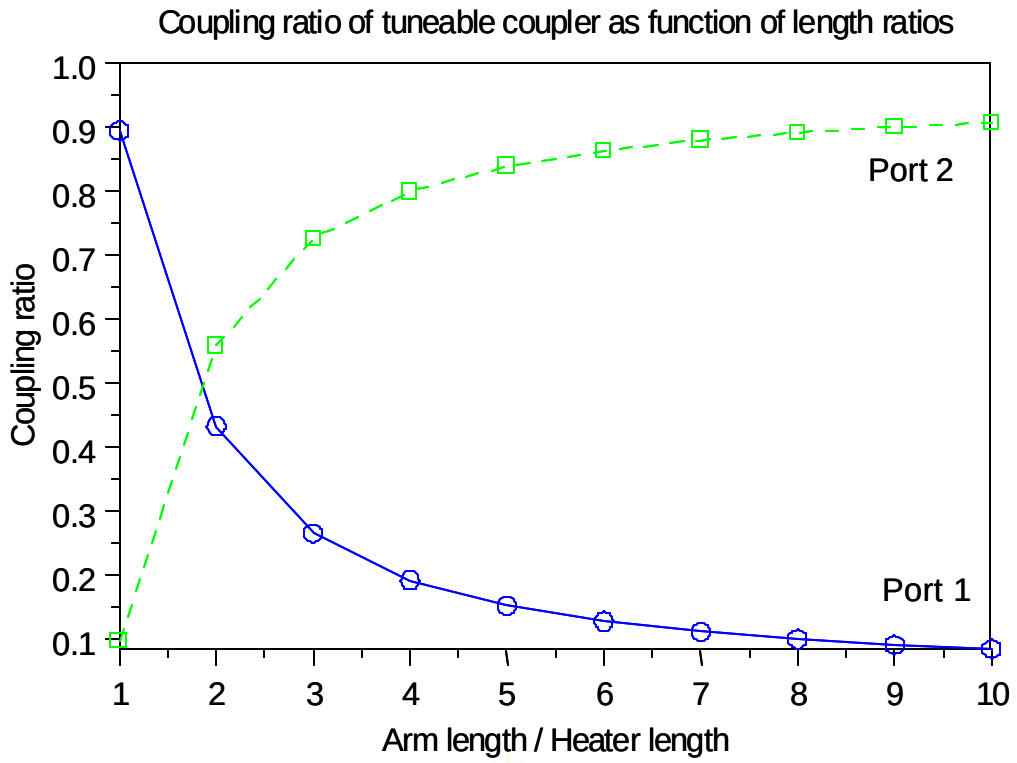

Figure 6.13: Output characteristic of symmetrical MZI vs. length ratios

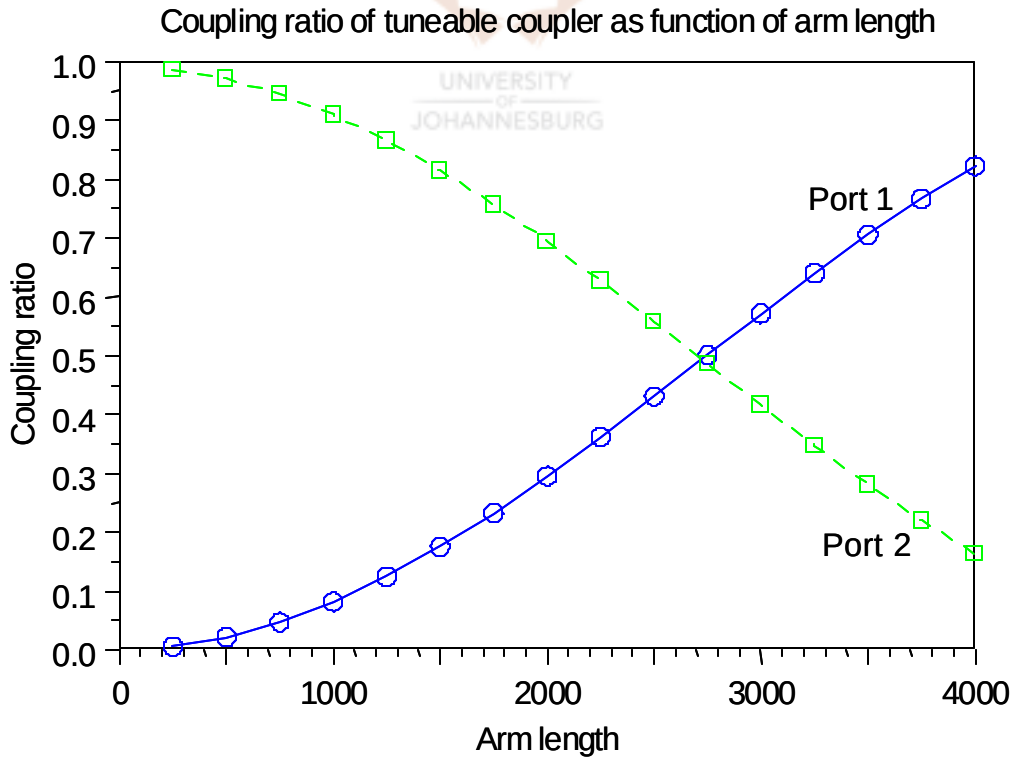

Figure 6.14: Output characteristic of symmetrical MZI vs. arm length

In conclusion, then, a tuneable coupler was designed with the following parameters (utilising Coupler 2 from section 5.2.2):

- Arm length = 2720μm
- Arm length/Heater length = 2

The final dimensions of the designed device are 6750μm × 27.4μm. The simulated result is presented in figure 6.15.

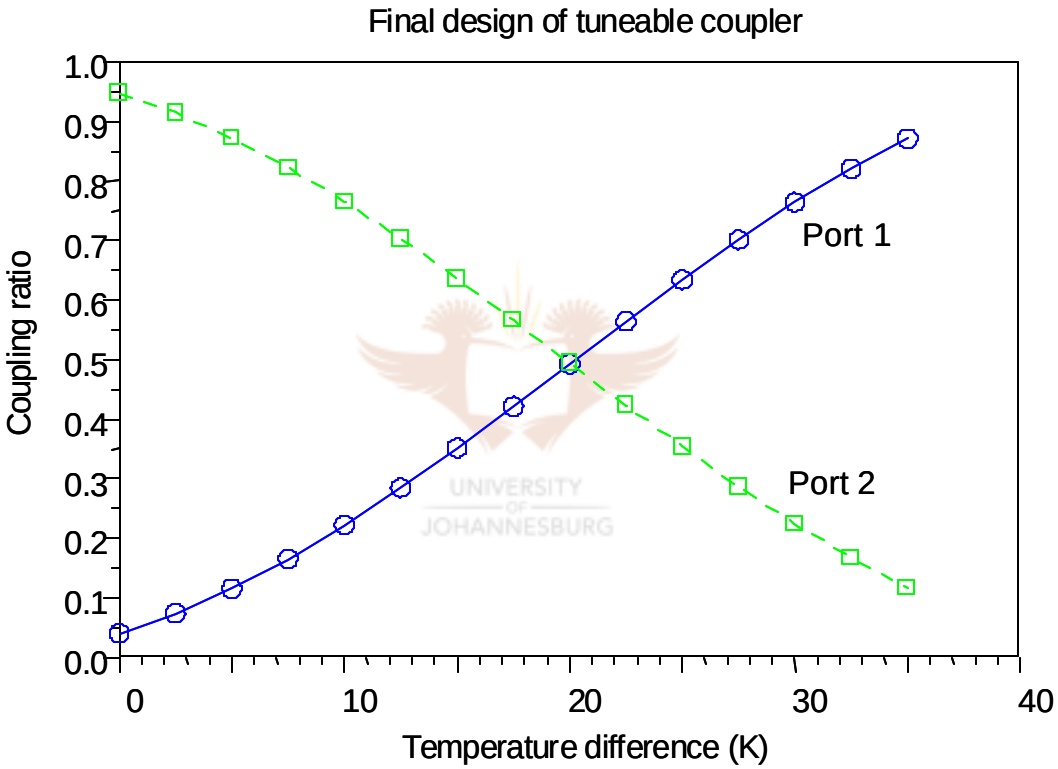

Figure 6.15: Output characteristic of symmetrical MZI tuneable coupler

6.6.2) Symmetrical MZI with MMI couplers

To obtain a tuneable coupler that is more robust to manufacturing variations, experimentation was done on MMI couplers to act as the splitting device. It was found that a push/pull geometry is needed to ensure a phase change of a large enough magnitude. A heater can be used on the one arm, while a cooler can be used on the other. The physical layout is presented in figure 6.16, while the BPM simulation results are presented in figure 6.17.

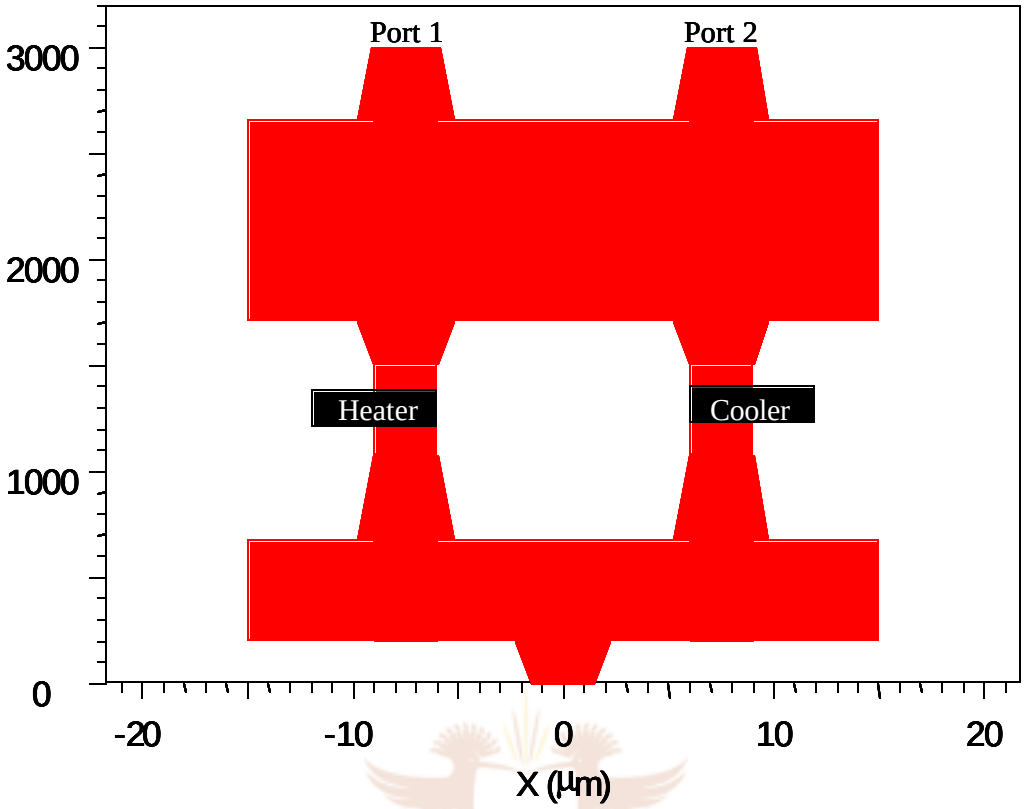

Figure 6.16: Physical layout of MMI tuneable coupler

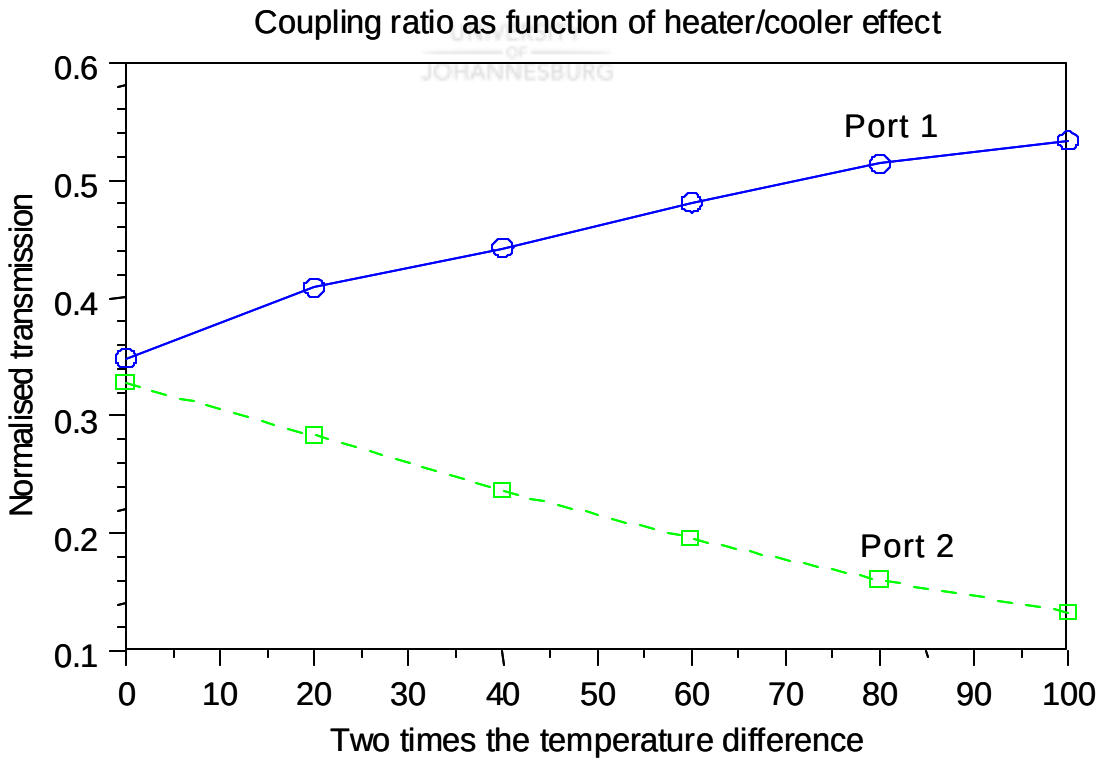

Figure 6.17: BPM results of tuneable MMI coupler

The coupler has a loss of 1.67dB, and is not as temperature sensitive as its directional coupler counterpart, but has the advantage of being almost twice as small. Optimisation work can be done in future on this design to improve the temperature sensitivity.

6.7) Simulation: MZI lattice filters

Simulations were performed in the BeamPROP environment to investigate the working of various lattice filters employing asymmetrical MZIs concatenated in a lattice structure.

For these designs, Coupler 1 in section 5.2.2 is used in order to reduce the size of the filters. Furthermore, we once again have to deal with the restriction on simulated angles. Because of the magnitude of the unit delay lengths (L_u) required to enforce certain free spectral ranges (FSRs), the angles tend to be large. Direct simulation is possible if the angles are made smaller, but the penalty is that the circuit becomes so large that simulation time is in excess of 1000 hours! A successful method of simulating these structures was developed. It involves the simulation of each coupling stage separately with BeamPROP, and a subsequent matrix multiplication performed in MATLAB.

6.7.1) Arbitrary example

Let the desired response be defined by $H_2(z) = 0.25[z^{-2} - 2z^{-1} + 1]$. The MATLAB program employing the recursion method was used to generate the required coupling ratios.

The following matrix indicates the calculated possible ratios:

	κ_2	κ_1	κ_0
Option1	0.14644660940673	0.50000000000000	0.14644660940673
Option2	0.14644660940673	0.50000000000000	0.14644660940673
Option3	0.85355339059327	0.50000000000000	0.85355339059327
Option4	0.85355339059327	0.50000000000000	0.85355339059327

It is important to note that these values were obtained by taking B_N as the input. This was done in order to ensure that equation 6.32 remains non-negative [6], and to achieve the symmetry that can be seen in the resulting coupling ratios.

In order to manufacture the shortest filter, option 3 or 4 may be chosen, i.e. the couplers must produce coupling ratios of 0.8536, 0.5000 and 0.8536 at a certain centre wavelength. If an FSR of say 10nm is required, the unit delay length is determined to be L_u = 159.86μm for a centre wavelength of 1.55μm.

In the next step, wavelength scans were performed for the specified couplers. It is important for the wavelength scan to cover the entire FSR. Finally, this data is loaded in a MATLAB program, where the matrix multiplication is executed. To facilitate rigorous scrutiny of the result, another MATLAB program is used to simulate the filter under ideal circumstances. In figure 6.18, the BPM simulation is presented whilst in figure 6.19, the required spectrum, ideal MZI and BPM-MZI are presented. From figure 6.19, it is clear that there was a slight wavelength shift to the left, which is due to the delay length not being defined to more than 2 decimal places after the comma in the simulation program, in order to incorporate manufacturing limitations.

The normalised bandwidths at the 3dB loss point were obtained as 0.63663μm for green (required), 0.62973μm for red (ideal), and 0.63405μm for blue (BPM). The transmission null for the BPM-MZI is shallower as is to be expected. This can be ascribed to an arbitrary mode that was launched which on closer inspection seemed to have an input power of slightly above 1 resulting in the favourable output. It is thus noted that subsequent simulations should be run with an input not exceeding 1.

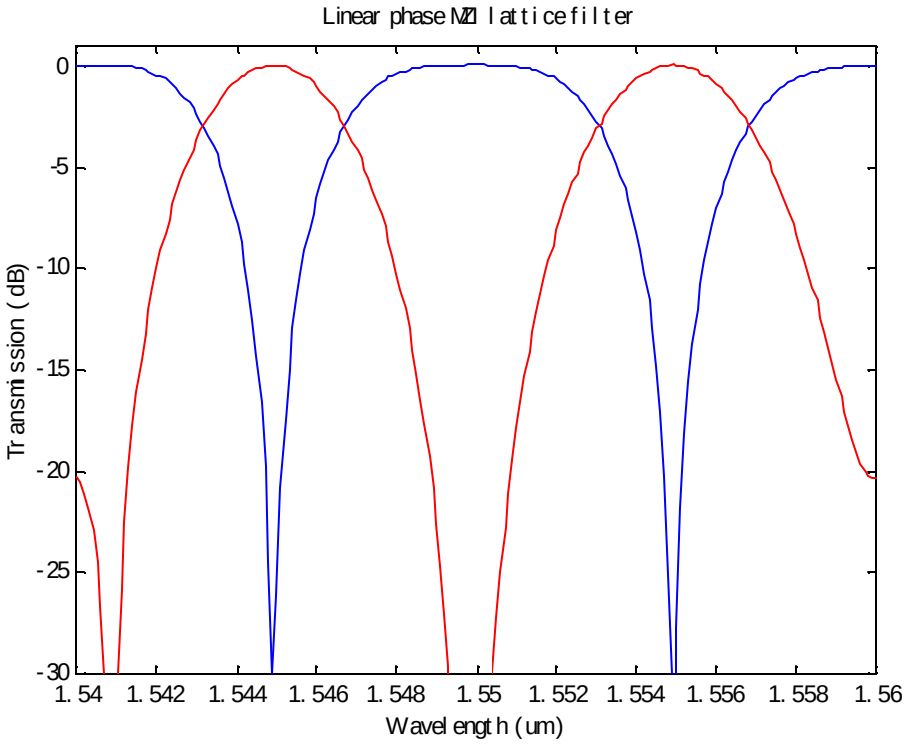

Figure 6.18: BPM simulation of arbitrary lattice filter

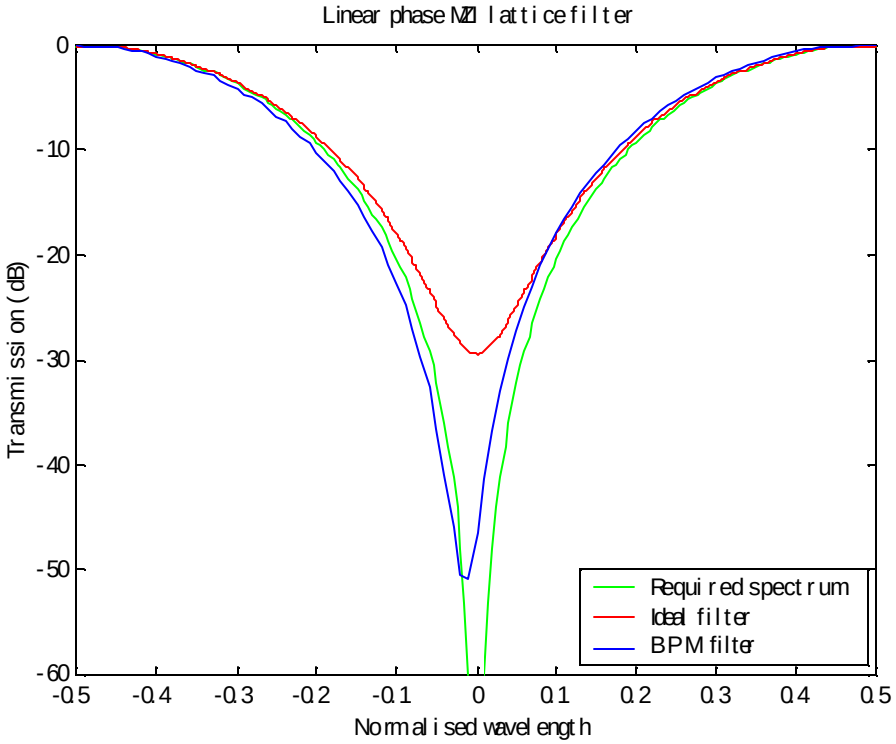

Figure 6.19: Comparison between required spectrum, ideal filter and BPM filter

6.7.2) Erbium gain equalisation filter

Gain equalisation filters (GEFs) are filters that compensate for the wavelength-dependent gain of optical amplifiers. These filters can be designed by using the negative of the gain spectrum as the required output spectrum.

After the wavelength region of interest has been selected (in this case, it will be between 1540nm and 1560nm), two approaches may be employed to yield the desired spectrum. First, we can let the region of interest represent half the FSR, and define the other half to be symmetrical about a normalised frequency $\nu = 0$. This approach will yield a linear phase filter. Secondly we can make the FSR slightly larger than the region of interest, and optimise the response for only the region of interest. In this case a mixed-phase filter is realised. In this discussion, the first approach will be used since the spectrum will be approximated by a 12^{th}-order least-squares fit.

In figure 6.20 the mirrored negative gain spectrum, least-squares fit curve, and an ideal MZI are presented. The MZI's coupling coefficients were generated with $B_N(z)$ as input to the MATLAB program.

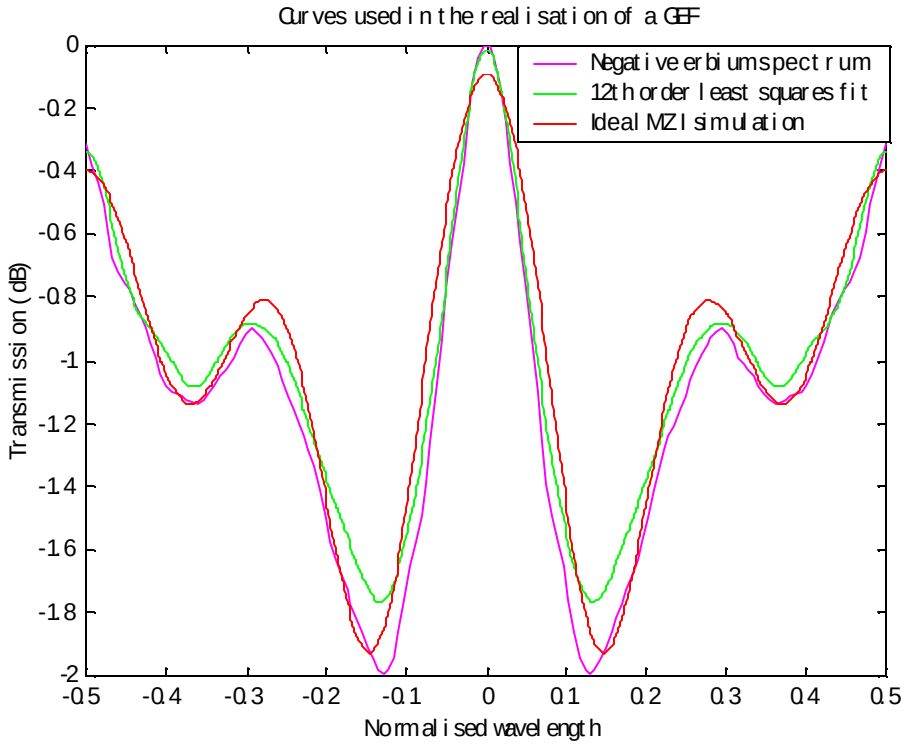

Figure 6.20: Curves used in the realisation of a GEF

It should be stressed that the least-squares fit is not optimised, resulting in the large ripple. Some of the coupling ratios obtained were very small, and were rounded off to zero. It is interesting to note that the rounded off coupling ratios, by deviating from the required green spectrum, approximate the actual erbium spectrum (the magenta spectrum) better than if the coupling ratios were exact. The resulting maximum deviation is 0.11dB.

BPM simulations using the coupling ratios of the above case did not yield useful results, mainly because the coupling ratio differences were too small. Even with the implementation of the generalised designs discussed in section 6.7.3, robustness of the design was not obtained. The coupling ratios can be improved by optimising the design for a smaller order (for example by using genetic algorithms), but this is beyond the scope of this thesis.

6.7.3) Generalised lattice filters

Generalisations can be introduced to the lattice filters as described in [7], which impact on the number of required stages and the filter's robustness to manufacturing variations. These generalisations are summarised as follows:

1) The delays in each stage can be integer multiples of the unit delay.
2) The relative delay can either be in the top or bottom arm of each MZI.
3) Non-integer multiples of the unit delay are used for optimisation, particularly when the wavelength dependence of the filter is important.

The generalisations are built into the transfer functions quite easily. The things to keep in mind are that the number of stages no longer defines the order of the filter, but rather the number of unit delays, as given by [4]

$$P = \sum_{n=1}^{N} |m_n| \tag{6.38}$$

where m_n describes the unit delays. Furthermore, a positive m_n denotes a delay in the upper arm, with the reverse being true for negative m_n. The number of paths for an N

stage filter is $Q = 2^N$. By using the same definitions as equation 6.14, the polynomials are given by

$$A_N^P(z) = \left[a_0 + a_1 z^{-m1} + \ldots + a_Q z^{-(m1+\cdots+mN)}\right] z^{mneg} \tag{6.39}$$

$$B_N^P(z) = \left[b_0 + b_1 z^{-m1} + \ldots + b_Q z^{-(m1+\cdots+mN)}\right] z^{mneg} \tag{6.40}$$

where a and b denote the transmission coefficients for each path, m_{neg} is the sum of the negative relative delays, the superscript P is the polynomial order and the subscript N is the number of the output stage. The multiplication by m_{neg} makes the z-transform causal. In order to incorporate the lattice filters in the recursion formulae described earlier, we can write them in matrix form as follows

$$\begin{bmatrix} X_n(z) \\ Y_n(z) \end{bmatrix} = \Phi_n(z) \begin{bmatrix} X_{n-1}(z) \\ Y_{n-1}(z) \end{bmatrix} \tag{6.41}$$

with for $m_n > 0$

$$\Phi_n(z) = \begin{bmatrix} c_n e^{-jf_n} z^{-m_n} & -js_n \\ -js_n e^{-jf_n} z^{-m_n} & c_n \end{bmatrix}$$

$$\Phi_n(z) = \begin{bmatrix} c_n e^{-jf_n} & -js_n z^{-m_n} \\ -js_n e^{-jf_n} & c_n z^{-m_n} \end{bmatrix} \quad \text{for } m_n < 0 \tag{6.42}$$

It is important to note that an additional phase shift of π should be added to the delay line every time there is a shift between positive and negative delay lengths to compensate for the lost coupling ratio shifts.

The generalisations presented in the above paragraphs add robustness to the designs where they are used to shorten the coupling lengths. Shorter coupling lengths are more resistant to manufacturing anomalies and are more wavelength insensitive.

6.8) Conclusions

MZIs and their filtering abilities have been discussed. The symmetrical MZI can be used in the manufacture of tuneable couplers, whereas asymmetrical MZIs can be used in the realisation of lattice filter structures.

Lattice filters are powerful optical filters because they allow for a systematic approach to filter synthesis, have a straightforward implementation, are both modular and regular (making expansion of the filter easy), have convenient numerical properties making them ideal for digital signal processing, and provide a natural model for many important physical and mathematical phenomena [8]. Successful lattice filters have been demonstrated, as well as simulations that not only regard the mathematical functions, but also the physical aspects that affect their functioning. Schemes for making them more robust were also discussed.

6.9) References

1) N. Takato, K. Jinguji, M. Yasu, H. Toba, M. Kawachi, "Silica-based single-mode waveguides on silicon and their application to guided-wave optical interferometers", Journal of Lightwave Technology, vol. 6, pp. 1003 – 1010, 1988.
2) C.K. Madsen, J.H. Zhoa, "Optical filter design and analysis: A signal processing approach", Wiley interscience, 2000.
3) K. Jinguji, M. Kawachi, "Synthesis of coherent two-port lattice-form optical delay-line circuit", Journal of Lightwave Technology, vol. 13, pp. 72 – 82, 1995.
4) B. Moshlehi, J. Goodman, M. Tur, H. Shaw, "Fiber-optic lattice signal processing", Proceedings of the IEEE, vol. 7, pp. 909 – 930, 1984.
5) M. Kawachi, "Silica waveguides on silicon and their application to integrated-optic components", Optical and Quantum Electronics, vol. 22, 99. 391 – 416, 1990.
6) S. Nordebo, "Advanced filter design: Minimum phase FIR filter design by spectral factorization", University of Karlskrona/Ronneby, www.its.hk-r.se/courses/sfd401/inlamningsuppgifter/uppg//.pdf.

7) Y. Li, C. Henry, “Silica-based optical integrated circuits”, IEE Proceedings in Optoelectronics, vol. 143, pp. 263 – 280, 1996.
8) B. Moslehi, J. Goodman, M. Tur, H. Shaw, “Fiber-optic lattice signal processing”, Proceedings of the IEEE, vol. 72, pp. 909 – 930, 1984.