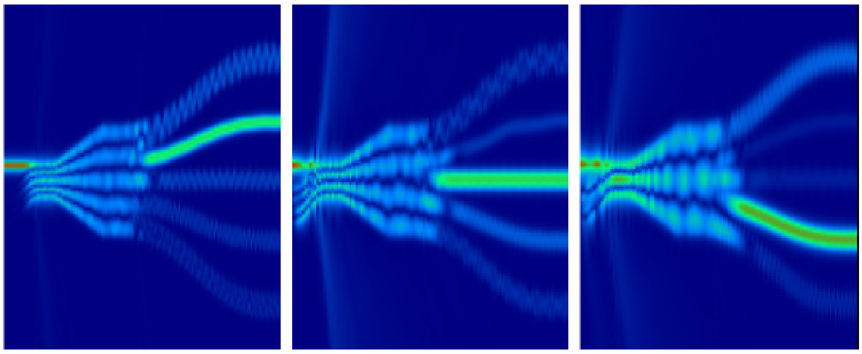
# Adiabatic Coarse Wavelength De/Multiplexing for Optical Systems



Andrew W. Molloy

A thesis submitted for the degree of
Bachelor of Science with Honours in Physics of
The Australian National University

October, 2004

# Declaration

This thesis is an account of research undertaken between February 2004 and October 2004 under the supervision of Prof. John Love and Dr Adrian Ankiewicz at The Research School of Physical Sciences and Engineering, Optical Science Group, The Australian National University, Canberra, Australia.

Except where acknowledged in the customary manner, the material presented in this thesis is, to the best of my knowledge, original and has not been submitted in whole or part for a degree in any university.

Andrew W. Molloy
October, 2004

# Acknowledgements

My interest and motivation in Photonics was cultivated in the coastal region of Newcastle. Much thanks goes to John Holdsworth for his guidance in these 3 developmental years. Moving to the cooler region of Canberra for my honours year I was welcomed into the physics community, being based in the Optical Science Group, RSPhysSE. My supervisors John Love and Adrian Ankiewicz need much thanks. Adrian always had time to give advice or to bounce an idea off, and also was a big help when i was learning BBV program. John thank you for sparing time in your busy schedule to make sure my projet was still on track. Also your constructive criticisms of my thesis writing was much appreciated.

My office buddy and friend Beata, made my honours journey enjoyable and fun. Some memorable moments being, the Yes Yes Yes, the high order soliton that didn't really exist, the two seminars you presented, and the submission of your first (well technically second) paper. Thanks for the conversations of my project ideas/problems, also i enjoyed returning the favour with respect your PhD project.

Thanks to all who helped check my thesis John, Beata, Sandra Sangeetha and Adrian.

The honours group could always find some time to celebrate the end of a weeks work, with Amber ale at the uni bar Friday afternoons. My soccer, indoor cricket and pool competition mates provided an essential honours escape.

Thanks for the support at home by my Dad and Sandra, although at times conflicts of lifestyle caused a few problems, we still lasted the year! Lastly my SR-250 motorbike needs thanks for having minimal running problems and safely getting me to and from ANU each day.

# Abstract

A novel approach to the simultaneous multiplexing or de-multiplexing of 3 wavelengths for coarse wavelength channel separation is based on the geometrical evolution and transformation of the modes of a single few-mode planar device. An input fundamental mode is transformed into a particular higher-order mode, depending on its wavelength, and then is transformed directly into the fundamental mode of a particular output port. The functionality of the device is quantified and its performance optimised through appropriate design. The device overall size is minimised, taking into account core spacing requirements for a fibre pigtailing, i.e. connecterisation of the input and output device arms to optical fiber.

# Contents

# List of Figures

# List of Tables

Chapter 1

# Project Summary

The foundation of this project was initially set with the project goals, consisting of:

- The investigation of the possibilities in making a 3-wavelength division multiplexing and demultiplexing device that implements a new geometrically based adiabatic mode evolution technique and has a reduced number of concatenated junctions to a previously developed device. A 3-wavelength division demultiplexing device has previously been developed with 4 concatenated adiabatic Y-junctions. The problem with concatenated devices is that the length can become impractical and with more junctions the 'excess loss' and 'crosstalk' increase in the device.

- Using a polymer based material with a refractive index difference of 0.027 is the design material, where the core and cladding refractive indies are 1.527 and 1.5 respectively. This represents the available polymer waveguide fabrication technology materials and is attractive due to the potential low cost of producing planar lightwave circuits.

- Tolerance determination of designed devices. Tolerance is an essential part of any design, as the fabricated device needs to operation to a predetermined standard. The tolerance of the device has to account for manufacturing error, aging, and temperature fluctuations all causing changes in design parameters.

My thesis is organised as follows:

Chapter 2 introduces Wavelength Division Multiplexing (WDM) systems, devices and issues of tolerance, where the focus is towards Passive Optical Network (PON)devices that operate in the Coarse WDM system and the adiabatic mode evolution technique. Why should this adiabatic device be investigated? It has the potential impact of reducing costs due to advances in polymer technology, although this impact can only occur if design tolerances are greater than the manufacturing error.

Modelling issue is addressed in chapter 3, where first the underlying physics of electromagnetic mode propagation is analysed and the modal effective index is calculated using the Scalar Wave Equation, i.e. an approximation of Maxwell's equations for the dielectric planar waveguide structures with weak-guidance properties, i.e. small refractive index variation between the cladding and the core. A (2D) planar slab waveguide model is used, as this has been shown to be accurate in related device studies, and it allows for a clear

understanding of the phenomena involved. Investigation of effective index variation is shown by characterising trends as a function of waveguide width for variable wavelength, mode order and refractive index difference parameters.

A key element of the adiabatic evolution method is explained through super-mode analysis. Finally the modelling section explores the possible methods of creating operating devices, shown are the obstacles of each method and also the enlightening of the best foreseen method used in the proposed devices. Fundamental elements of the adiabatic device such as n-way asymmetric junctions plus tapers are studied in chapter 4.

The modelled components are combined in chapter 5, to produce operating devices. Demultiplexing and multiplexing properties are highlighted. The devices include an improvement in a 2-wavelength demultiplexer, a new 3-wavelength de/multipler and the possibilities in development of add/drop multiplexers. All developed devices operate with approximate adiabatic mode evolution and have no interference, reflection or mode coupling effects. These devices are compared to earlier adiabatic devices. Then in chapter 6 tolerance is investigated, to determine the feasibility of physically making such demultiplexing devices.

Chapter 2

# Introduction

Coarse Wavelength Division Multiplexing (CWDM) technology is proving to be a cost-effective solution for higher information capacity metropolitan and local area networks. CWDM devices can use interference, reflection, mode coupling or described here the new technique of interest of adiabatic mode evolution to multiplex and demultiplex wavelengths. While designing such a device, it is important to consider if the device will operate within a particular tolerance range. Tolerance factors dictated here are by source wavelength drifts, source bandwidth, and manufacturing errors.

## 2.1 Wavelength Division Multiplexing Systems

In telecommunication networks the challenge is to increase the information capacity at a minimal cost or in a cost-effective way. An optical fibre allows for high bit rates and therefore higher capacity communication solutions. Wavelength Division Multiplexing (WDM) is a technique of Frequency Division Multiplexing (FDM) applied at optical frequencies i.e. terahertz, usually referred to by wavelength; for example 193.3 THz converts to 1550nm. WDM allows an optical fibre to have multiple channels propagating simultaneously. A channel is a collection of wavelengths about a centre wavelength, or a small bandwidth about a centre frequency. In a basic point-to-point link multiplexing occurs when multiple inputs, each operating at a different wavelength, are transformed into a single output with the multiple wavelengths being superimposed. Demultiplexing reverses the multiplexing transformation such that the multiple wavelengths are separated into independent outputs.

There are limits to how much information can be sent down an optical fibre. The parameters to be considered in an optical network are the laser sources or wavelength band, wavelength separation, bit rate, type of fibre/s in the network and the associated attenuation and dispersion characteristics, link length, amplification requirements, and detector characteristics. The parameters affect the quality of service such as crosstalk and signal-to-noise ratio. The telecommunication problem has been investigated to the extent that standards have been established for WDM systems. A characteristic for all WDM systems is transparency to protocol[1], this gives the WDM systems the ability to be impartial to transmitted protocols such as SONET/SDH [Synchronous Optical Network/Synchronous Digital Hierarchy], Gigabit Ethernet, fibre channel, and ATM [Asynchronous Transmission Mode].

WDM systems can be characterised into two regimes including Dense WDM (DWDM) and Coarse WDM (CWDM). DWDM system was developed for long haul communications where wavelengths are spaced as close together as possible with high bit rates. Compacting of wavelengths is required to utilise the limited gain spectrum of Erbium Doped Fibre Amplifiers (EDFA). Amplification is used to compensate for link losses. EDFA gain spectrum limits the simultaneous amplification to 1540-1570nm wavelength range or the C-band and also long wavelength EDFA amplifiers extend amplification to 1570-1605nm. CWDM is used for low/medium capacity and short link length systems, typically requiring no amplification.

Operation wavelengths for WDM systems depends on dispersion effects, gain spectrums, laser sources and fibre loss mechanisms of absorption, scattering and bend loss. Bend loss can be avoided in optical fibre cables but it can be a significant source of loss in integrated optical components as tight bends are incorporated in compact designs. Telecommunication windows have been developed around the properties of silica fibre and optical laser sources. Nomenclature used for telecommunication windows are original (1260-1360nm), extended (1360-1460nm), short (1460-1530nm), centre (1530-1565nm) and long (1565-1625nm) bands as in Figure 2.1 [1]. There are also eight popular CWDM channels near 778 - 864nm (in accordance with IEEE 802.3z standard)[2]. The 'water peak' from G.652 type fibre is present in the E-Band, i.e. an absorption peak caused by $OH^-$ ions that are embedded in silica fibre in the manufacturing process. With new advances in fibre fabrication, the water peak can be substantially eliminated by reducing hydrogen in the process of manufacturing a fibre; this has led to a new standard of fibre ITU-T [International Telecommunication Union] G.652c fibre.

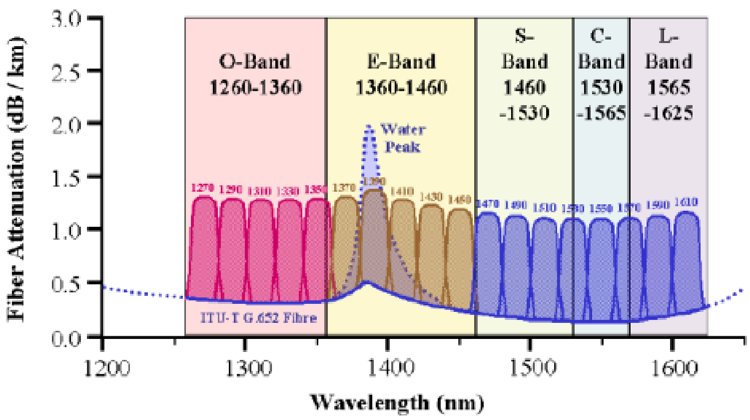



**Figure 2.1:** Telecommunication bands and the ITU CWDM grid are superimposed over silica fibre attenuation of old and new standard fibres. The 'water peak' at 1380nm reflects older optical fibre where the $OH^-$ ion absorption peak was caused by the ions being fused into the fibre during manufacture process. New fibres G.652c have eliminated the water peak down to 0.5 dB/km which has effectively made the E-Band operational. The C-Band bandwidth is small and reflects the narrow operating spectrum of EDFA's

[RBN inc, Characteristics of CWDM. Roots, current status and future opportunities, 2002] Difference between DWDM and CWDM can be examined through major components such as transmitters, receivers, filter/multiplexing, and amplification devices [1], and are discussed below.

Optical transmitters are laser sources that operate with a narrow bandwidth about a centre wavelength. To transmit information the laser source is modulated at a particular bit rate. Information sent per channel is determined by the bit rate. Bit rate is dispersion limited, such that inter-symbol interference does not occur, i.e. where two adjacent bits '0' or '1' become indistinguishable at the receiving end of the link. The bit rate also limits the wavelength separation by the induced chirp through source modulation, i.e. the bandwidth of the laser source will be increased through modulation. Thus adjacent channel separation must be adjusted to avoid frequency/wavelength space overlap or crosstalk. CWDM lasers operate up to 2.5Gbits/s while DWDM lasers typically operate at up to 10 Gbit/s, although this comes at the expense of a stabilised laser. DWDM lasers costs about four times that of a CWDM laser and also require higher operational costs with heat control and output power differences.

Receivers are photodiode detectors that convert the information from the optical domain back to electrical. Photodiodes operate over a specific optical spectrum range that is dependent on the bandgap of the detector. P-Intrinsic-N (PIN) photodiode and Avalanche PhotoDiode (APD) detectors are used. For passive WDM systems detecting of transmitted channels could require broadband detectors or multiple detector that collectively detect over the spectral range of the CWDM spectrum, while DWDM detectors only require one general detector.

CWDM filters/Multiplexers/Demultiplexers are unidirectional or bi-directional devices and in most cases are quicker to manufacture. This leads to reduced costs, relative to DWDM devices. This reduced cost is due to tolerance requirements being relaxed with CWDM spectral filters responses.

Loss due to scattering, bend loss, absorption, insertion loss of optical components in an optical network needs to be balanced by gain, such as laser input power and amplification. CWDM is a short-haul system and does not need amplification. It also allows total utilisation of the telecommunication bands (O, E, S, C and L) while DWDM is a long haul application that typically uses EDFA's to amplify the multiple channels simultaneously.

| | **Coarse WDM** (includes WWDM) | **WDM** | **DWDM** (includes ultra dense WDM) |
|---|---|---|---|
| Channel Spacings | Large, from 1.6 nm (200 GHz) to 25 nm | 1310 nm lasers used in conjunction with 1550 nm lasers | Small, 200 GHz and less |
| Number of bands used | O,E,S,C and L | O and C | C and L |
| Cost per channel | Low | Low | High |
| Number of channels delivered | 17-18 at most | 2 | Hundreds of channels possible |
| Best Application | Short-haul, Metro | PON | Long-haul |

**Table 2.1**: Types of wavelength division multiplexing.

DWDM is primarily used for long-haul telecommunication links shown in Table 2.1. In metropolitan area networks the system parameters are changed, particularly the distance of the link/network and the capacity level. In such a case the cost effective approach is to use CWDM.

An applications of CWDM is fiber-to-the-X (FTTX), where X can be Home, Building or Curb. FTTH access systems offer high-bandwidth, dedicated and new services to residential and business subscribers. Passive Optical Network devices (PON) devices can be used instead of active components that require power sources. The advantage of FTTH PON is the fact that they use purely optical passive components that can withstand severe and demanding environment conditions without the need to consume electrical energy [3]. An example of proposed CWDM services operating in four optical channels are: 1310nm for data/voice upstream signals, 1490nm data/voice down stream signals, 1550nm for video signals and 1625nm for remote networks fault detection by OTDR [Optical Time Domain Reflectrometry] technique.

## 2.2 Demultiplexing Device Operations

How do devices operate to enable multiplexing and demultiplexing? Spectral filters are used to control the behaviour of particular spectral regions. For example a demultiplexing filter controls the wavelengths that propagate in the multiple output waveguides. Devices function with one or more of the following techniques: interference, reflection, mode coupling, and adiabatic mode evolution. These techniques with some examples are discussed below.

Interference utilises the constructive and destructive nature of light when a coherent source is separated then combined together with a phase difference. If the phase difference is 0, $n\pi$ where $n = \pm 2, \pm 4, \pm 6...$ there will be maximal constructive interference, while null power results occur at phase differences of $m\pi$ where $m = \pm 1, \pm 3, \pm 5...$ due to total destructive interference. Examples of interference based devices are Mach-Zehnder, Arrayed Waveguide Gratings(AWG) and MultiMode Interference (MMI).

The Mach-Zehnder interferometer has the two or more input wavelengths 50:50 (or 3dB) coupled into two paths by a beam splitter, e.g. a symmetric Y-junction. The two paths have different lengths and when the paths are recombined into a coupler configuration, i.e. the two joining identical waveguides are slightly separated, there is a phase difference in two coupler inputs. Due to the path difference each combinations of anti-symmetric and symmetric super-modes are excited in the coupler, for each of the two coupler inputs. For a $\pi$ phase difference the super-modes will interfere leaving power in only one output arm, this phase oscillation continues with coupler length. The coupler length required for the power to be converted from one arm to the other is wavelength dependant and is know as the coupling length. Thus for demultiplexing to occur it is also possible to use the Mach-Zehnder filter to interleave between even and odd frequency channels [4](increment spacing is only uniform in the frequency domain) for an appropriately selected coupler length .

Arrayed Waveguide Gratings utilise path difference caused by the light propagating along multiple parallel waveguides of different lengths. The path difference creates a diffraction grating effect (diffraction orders) in the output medium and thus different wavelengths can be focused into separate waveguide outputs.

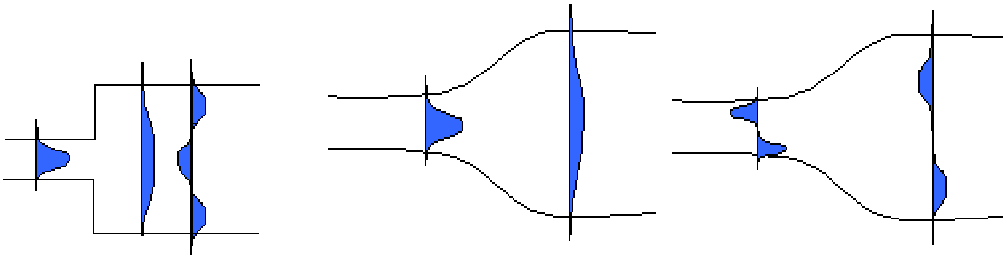
Reflection in a fibre Bragg grating is caused by a periodic or quasi-periodic series of refractive index variations in the translational direction along the fibre. This periodic change in refractive index causes reflections at the Bragg wavelength, i.e. the wavelength that the reflections constructively interfere, the phase matching condition is $\lambda_0 = 2n_{eff}\Lambda$, where $n_{eff}$ is the average effective index of the grating and $\Lambda$ is the Bragg period of refractive index variations [5]. The reflection can also be thought of as constructive interference or mode coupling. Bragg grating apodisation function represents the change in refractive index in the spectral domain. The reflected spectrum is determined by making a Fourier transform of the spatial domain to the frequency/wavelength domain.

Acousto-optic filters are tuneable filters where the acoustic frequencies transmitted through a glass material create high and low density regions inside the glass material [4]. This sets up a tunable Bragg grating or interference filters properties.

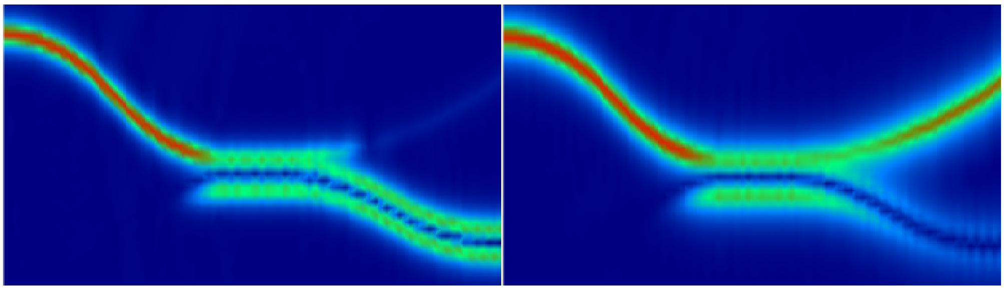
Mode coupling occurs when the partial or total power from one mode is transferred into another propagating mode/s, and is used in WDM devices by coupling between two parallel waveguide/fibre cores [4][6]. Mode coupling can also be created by abrupt changes in a waveguide width or refractive index, for example Multi-Mode Interference (MMI) devices rely on multiple mode excitation for interference to take place. In Figure 2.2(a) a sudden translational change in the waveguide width causes the fundamental mode to excite higher order modes, e.g. $\Psi_0$ and $\Psi_2$, in the wider waveguide section. In such a symmetric geometry only even modes are excited in the larger waveguide structure, while an asymmetric geometry can excite even and odd modes.

If a waveguide cross-section is gradually increased or decreased significantly slowly [6] with propagation distance the mode coupling phenomenon will become insignificant. In such a process there is virtually no transfer of power from one local mode to another and it is referred to as an approximately adiabatic taper. This allows the propagating mode to conserve power, although its mode field diameter, modal propagation constant and intensity distribution will be changed. Figure 2.2(b) shows a translational variation in a waveguide taper, the gradual change in waveguide width essentially prevents mode coupling. Thus adiabatic propagation results depicted in this adiabatic taper are the odd fundamental/$\Psi_0$ and even $\Psi_1$ modes.

The wavelength switching ability of two adiabatic asymmetric Y-junctions is shown in Figure 2.2(c). The narrow input arm supports the fundamental mode. The input arm is met by a phantom arm, i.e. an arm that has no input source and has a larger waveguide width relative to the input arm. The combined width creates the linking waveguide between the two Y-junctions, also referred to as the stem. The input wavelengths modal effective indies are closely matched to the $\Psi_1$ stem mode. This match converts the input wavelength modes into the stem $\Psi_1$ mode. The left 850nm case has the $\Psi_1$ mode evolved



(a) Mode coupling (b) Adiabatic evolution of $\Psi_0$ and $\Psi_1$



(c) Adiabatic evolution through a junction demultiplexing 850nm and 1550nm in the left and right respective field plots

**Figure 2.2:** a)Mode coupling, b)adiabatic mode evolution, and c) basic two-wavelength demultiplexer.

into the wide bottom output arm. While in the right 1550nm case, the $\Psi_0$ is evolved into on the narrow top output arm.

## 2.3 Adiabatic CWDM Device

Of particular interest for this project is this new adiabatic mode evolution technique. The simplest example of a planar slab device that demultiplexes two wavelengths shown in 2.2(c) [7]. In general the demultiplexing device will have $n$ wavelengths entering the input, all assumed to be propagating in $\Psi_0$ mode. Then each wavelength is evolved into separate $n$ output ports/arms. The input-stem junction is designed to adiabatically evolve each $n$ fundamental inputs into one $nth$ higher order stem modes, e.g. $\Psi_1, \Psi_2$ or$\Psi_3 ... \Psi_n$. To avoid a step function change in waveguide width at the Input-Stem junction, 'phantom' input/s are required such that the sum of the input and phantom input widths equals the stem width. Similarly the Stem-Output junction requires that the sum of output arm widths equal the stem width.

When demultiplexing two wavelengths intuitively one would expect two output arms. This may or may not be the case, as the device is driven by modal effective index matching [8]-[9]. Increased control and flexibility may be required, leading to additional dummy arms that acts to maintain the adiabatic process and also reduce restrictions on the operational arm widths. Ideally the adiabatically evolved output modes are single moded. Reversing the propagation direction of the light, the device should act to multiplex by the reciprocity law. Note that the phantom input width will have an effect on the mode evolution. In such a case reciprocity needs to be checked. These ideas and techniques are developed further throughout this thesis.

In this new type of device super-modes evolve adiabatically through an asymmetric junction. The junction may consist of a Y or n-way splitter with n output arms. The key concept for the de/multiplexing operation is that the input super-mode is required to evolve adiabatically into the an output supermode that has power is concentrated to one propagating mode, in one constituent output arm.

Adiabatic choice comes about by the match of the effective index in the input and output junction propagating super-modes. Approximation of waveguide widths in the input and output structure can be made by assuming that the super-mode's effective index is the same as the mode propagating in an isolated waveguide. This is approximate as the super-mode effective index in regions of tapered n-waveguides, e.g. 1 to 4-waveguide taper, have slightly less effective index values relative to an the isolated waveguide model. The super-mode is influenced by the whole cross-sectional geometry of the junction.

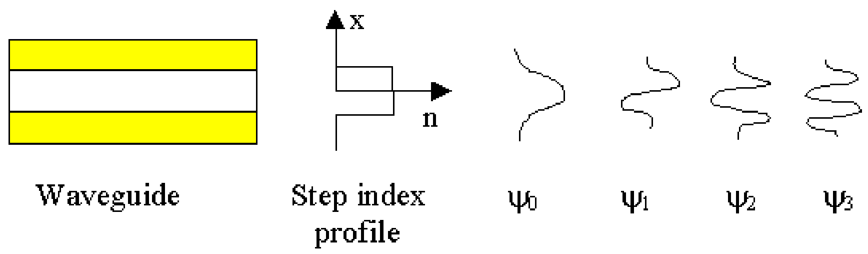
Modal effective index is $(n_e) = \beta/k$ where $\beta$ is the propagation constant for a discrete propagating mode and $k = 2\pi/\lambda$ where $\lambda$ is the free space centre channel wavelength. Bound modes exist when the modal effective index is between the refractive index of the core and the cladding. Waveguides in this thesis have step index profile, shown in Figure 2.3(a) for an isolated planar slab waveguide structure. Leaky modes (by radiation) exist when $n_e$ is less then the cladding refractive index.

Effective indies of propagating modes are influenced by the control parameters of width (input, phantom inputs, stem and output arms), wavelength, refractive index profile. The control parameters are tuned to drive wavelengths to particular modes and output arms. Examples are shown in Figure 2.3(b).

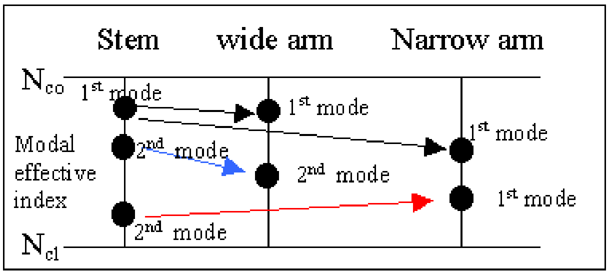
An asymmetric Y junction has been developed to demultiplex 1550 and 1300 nm light by, and the simulation has been already presented in Figure 2.4 [8]. The approach is based solely on the geometrical design of the device and approximately adiabatic evolution of modes through the device. Also assumed is a weakly-guiding step index profile. The input $\Psi_0$ mode is evolved into the $\Psi_1$ mode in the stem and then back into the $\Psi_0$ mode in the narrow output arm. The angle between the output arms is small enough for the device to be approximately adiabatic and have negligible excess loss.

Extension to the two-wavelength demultiplexer is to demultiplex three wavelengths and this can be achieved by concatenating asymmetric Y-junctions, shown in Figure 2.3 [9]. The device was modelled for a 0.04 difference in refractive indies based on $As_2S_3$ type of glass, where the longest wavelength is placed in the fundamental mode in the first narrow output, then the two shorter wavelengths are evolved into the $\Psi_1$ mode of the first wide arm, this process is then continued until all wavelengths are in the fundamental modes.

Asymmetric Y-junctions have been shown to possess the ability to demultiplex wavelengths. Moore's Law indicates that devices tend to become smaller or have increased operation capabilities as the device advances, so how can this be applied to the adiabatic




(a) The planar slab waveguides with the characteristic of having a step refractive index profile at the core and cladding interface, the core having the higher refractive index. Low-order eigenmodes propagating along waveguide are of interest here and examples are $\Psi_0, \Psi_1, \Psi_2$ and $\Psi_3$. The eigenmodes exponentially decay in the cladding while are oscillatory in the core.




(b) Adiabatic mode evolution effect examples: Where the fundamental mode is propagating in the stem the effective index match to the output arm will always choose the widest output arm. Effective index increases with width. When the second mode propagates in the stem, two things can happen depending on the wavelength (assume an appropriate selection of narrow and wide arm widths) ; the closest match of effective index would be the fundamental in the narrow arm for the longer wavelength and the second mode in the wide are for the shorter wavelength.

**Figure 2.3**: a) Waveguide properties, b) Modal manipulation examples

de/multiplexing? Increase in the number of operational channels per device and/or reducing the length of device by directly evolving the stem modes into fundamental output modes, i.e. instead of requiring additional Y-junctions to convert modes into the fundamental modes. An attempt to show potential increased device capabilities, while also reducing device size is undertaken in this thesis.

Characteristics of adiabatic process for multiple output channels and modes have been highlighted in [10]. Asymmetric adiabatic multiprong mode-multiplexed systems with up to four modal channels are analysed. Devices can selectively drive/sense the individual modes of up to four mode waveguide with extinction ratios of up to a -25db and negligible excess loss. Similar modal switching devices have lengths of 10, 15, and 50 mm for respective 2, 3, and 4 mode channel splitters. From these dimensions, the focus for this thesis was expected to be on 2- and 3- output arm structures, as devices with 4 modal output arms becomes unacceptably long.

The Local mode at the input are even and odd mode in a square core waveguide; at the output the local modes are the modes that are the modes of an asymmetric coupler, which are almost exactly the eigenfields of the 2 constituent guides in isolation propagating single-moded, i.e. sufficiently different and well separated. Selection of bend is also highlighted

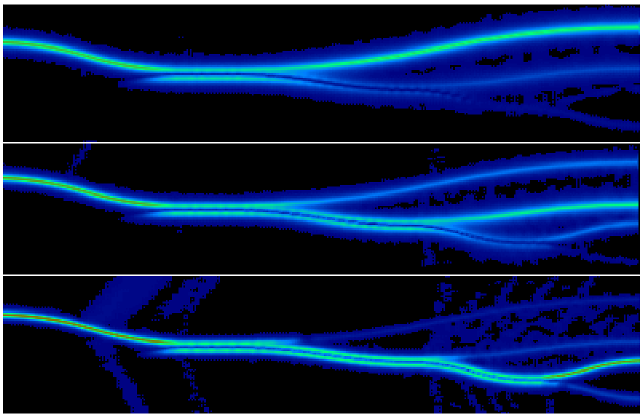


**Figure 2.4:** From left to right the device sections are the input Y-junction, stem that connects to the first output Y-junction, the second stem then meet the second output Y-junction followed by the last stem and Y-juncition. The input $\Psi_0$ mode is evolved into the $\Psi_1$ stem mode. For the 1550nm channel the stem mode evolves back into the $\Psi_0$ mode in the narrow output arm. The smaller wavelengths continue in the $\Psi_1$ mode along the second stem and are met by another Y-junction where 1300nm evolves into the narrow output arm in the $\Psi_0$ mode. The process is repeated for the 850nm wavelength at the final Y-junction. Investigating the transitions between sections highlights the adiabatic process with respect to the super mode.

in the mode splitter device where the output/input arm bends are important, with lower order modes being more efficient when less deviated.

Other adiabatic devices have also been made but not for WDM applications. These include velocity couplers, directional couplers, mode converters and reduced bend loss [13]-[18].

## 2.4 Tolerance

Tolerance is an important part of device design, as it will indicate if the device will operate to an established standard in changing conditions, e.g. temperature. Another important aspect involves fabricating the device, this requires a process has a better resolution the the Critical Dimension (CD) or a greater accuracy then the device tolerance. Issues at the heart of WDM devices tolerance are crosstalk and insertion loss.

Crosstalk can be defined as homodyne or heterodyne. Homodyne is where two or more channels produce signals at the same wavelength. This results in interference of signals over the bit period leading to bit error. Examples of homodyne crosstalk could be laser source bandwidth overlapping or non-linear effects redistributing power into another channel.

Heterodyne crosstalk is induced by an adjacent/neighbouring channel where the detector's bandwidth is not narrow enough to detect power for only one channel. The total power reading of the detector is then the sum of the power of the central channel plus partial power from adjacent channels. For the case of adjacent channels being 'on' or a 'mark 1' and the central channel being 'off' or a zero, a bit error can be made in detecting the central channel signal caused by the adjacent channels.

Insertion Loss of components and fibre in an optical network collectively add to give the total link loss. This total loss in the system must at least match the gain in the system. For CWDM networks amplification is avoided as it adds costs to the network and also limits the possible information capacity of the network. Insertion loss is the reciprocal ratio of the output when device is inserted into link to the power received without device, and is expressed in dB. The loss may originate from the material used in the device at absorption peaks or generation of significant reflections caused by a large change in refractive index (usually insignificating), mode miss match between a connected optical fibre could cause large loss if an error is made in allignment. Also insertion loss may be influenced by manufacturing errors that impinge on the foundations of device operation, in this case adiabatic mode evolution.

### 2.4.1 Manufacturing Tolerance

Planar slab waveguide devices can be made in a number of fabrication process, where a mask of a device is transformed onto a silica or polymer based material. For example Planar Lightwave Circuits (PLC) are made by plasma enhanced chemical deposition and reactive ion etching, or direct writing [12]. The manufacturing accuracy is limited by the resolution in the fabrication process.

### 2.4.2 Wavelength Tolerances

The transmitted optical channel operation for both CWDM and DWDM are depicted in Figure 2.5 [1]. Highlighted is the broad spectrum that a CWDM source can operate in without causing crosstalk to adjacent channels. Uncooled laser operation can have wavelength fluctuations of $\pm$4.2nm about the boundary wavelengths of the source bandwidth. DWDM sources on the other hand have narrow bandwidth operation, i.e less then 0.8nm, and are stabilised by temperature control. The impact on design for the CWDM device is simply that the pass band spectrum for a demultiplexing device must account for all possible wavelengths of a channel. Features are summerised in table 2.2.

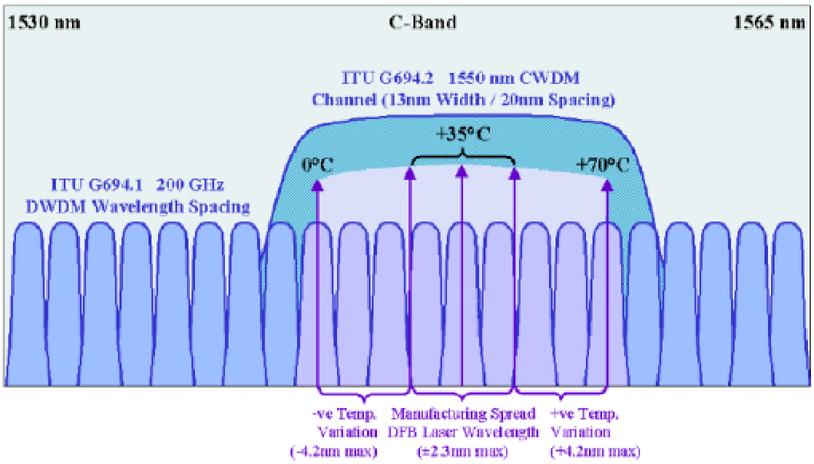



**Figure 2.5**: CWDM Vs DWDM - Laser source tolerance

| **Features** | **DWDM** | **CWDM** |
|---|---|---|
| Lasers Drifts | Cooled to stabilize +/- 0.06nm/$^0$C | Uncooled (wavelength drifts) +/- 3nm |
| Wavelength Spacing | 0.8, 0.4, 0.2nm | 20nm |
| ITU Grid | G694.1 | G694.2 |
| Bit Rate (Modulation effects) | Up to 10 Gbits/s | Up to 2.5 Gbits/s |
| Dispersion Problem | High with long-haul links | Low for short-haul, dispersion limited depending on fibre SM-28 or DSF |
| Amplification | Required | Not Required |
| Cost | High Devices and maintenance | typically much lower then DWDM |

**Table 2.2**: Summary of Tolerance Issues.

Chapter 3

# Modelling

This chapter gives an overview of the Scalar Wave Equation with weak-guidance approximations [6]. Applying the Scalar Wave Equation to a planar slab waveguide, solutions for effective indies can be solved numerically. This then enable the effective index to be characterised for varied design parameters. Super-modes are also investigated in symmetric Y-junctions, where the basic physical principles described can be extended to proposed devices this thesis. Finally the modal manipulation methods are devised based on the effective index characteristics.

## 3.1 Effective Index

Effective index is an important design consideration for the proposed devices, discussed in detail in chapter 5, where approximate effective index match drives the adiabatic process. The effective index, is shown to be affected by the operating wavelength, mode, waveguide component widths and refractive index. To characterise these effects, numerical solutions to the waveguide parameters are calculated, as there is no simple analytical solution. Using these characteristics, suitable dimensions for input, stem and output arms can be determined to produce a device that has fundamental modes propagating in the input and outputs.

### 3.1.1 Waveguide Analysis

The characterisation of both the fundamental and low-order multi-modes that propagate along an optical dielectric waveguide are determined using Maxwell's equations. In the scope of this thesis, Maxwell's equations can be simplified for dielectric material with only a small refractive index variance, i.e. the weak guidance approximation. The refractive index difference of 0.027 used in the proposed devices corresponds to a 1.8% relative difference, thus it is small. This approximation then leads to the scalar wave equation. Conditions for simplification are described below.

#### Scalar Wave Equation

Maxwell's equations are introduced below and are followed by the simplifications.

$$\nabla \times \mathbf{E} = -\mu \frac{\partial \mathbf{H}}{\partial t} \quad (3.1)$$

$$\nabla \times \mathbf{H} = \sigma \mathbf{E} + \epsilon \frac{\partial \mathbf{E}}{\partial t} \quad (3.2)$$

$$\nabla \cdot \mathbf{E} = \rho / \epsilon \quad (3.3)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (3.4)$$

In the above equations **E** is electric field vector, **H** is magnetic field vector, **B** is the magnetic flux density, $\mu$ is the dielectric permeability, $\epsilon$ is the dielectric constant, $\sigma$ is the conductivity and $\rho$ is the charge density.

In dielectric material there are no charge or currents (i.e source free) and therefore the charge density ($\rho$) and conductivity ($\sigma$) are assumed to equal zero. Also the dielectric permeability $\mu$ can be approximated to equaling the permeability of free space thus $\mu = \mu_0$.

The dielectric constant $\epsilon$ varies depending on the material and thus changes with the waveguide profile. Refractive index 'n(x,y,z)' profiles are used to describe waveguides structures thus the wave equation should have a refractive index term. Refractive index (n) squared equals the relative dielectric constant ($\epsilon_r$) or the dielectric constant $\epsilon = \epsilon_r \epsilon_0 = n^2 \epsilon_0$ where $\epsilon_0$ is the permittivity of free space.

Electric and magnetic fields are of interest in the waveguide analysis. The electric (**E**) and magnetic (**H**) are time dependant fields vectors, by confining attention to monochromatic waves (i.e. single frequency waves) the **E** and **H** fields can be written in the following cartesian coordinate form:

$$\mathbf{E}(x,y,z,t) = \mathbf{E}(x,y,z)\exp(-i\omega t) \quad (3.5)$$

$$\mathbf{H}(x,y,z,t) = \mathbf{H}(x,y,z)\exp(-i\omega t) \quad (3.6)$$

where $\omega$ is the radian frequency or $\omega = kc$ where the wavenumber $k = \frac{2\pi}{\lambda}$ and $\lambda$ is the free space wavelength.

With the above conditions the Maxwell equations (3.1)-(3.3) can be simplified. The (3.3) becomes equal to zero leaving the two coupled equations of (3.1) and (3.2) to become the following respective equations:

$$\nabla \times \mathbf{E} = \imath k c \mu_0 \mathbf{H} \quad (3.7)$$

$$\nabla \times \mathbf{H} = -\imath k c \epsilon_0 n^2 \mathbf{E} \quad (3.8)$$

Waveguides usually have the same refractive index cross section in the propagating direction (z), this is known as translationally invariant. In the translationally invariant case the electric and magnetic fields can be expressed as the product of the longitudinal and transverse dependance, shown here in cartesian coordinates

$$\mathbf{E} = \mathbf{E}(x,y)_t \exp(i\beta z - i\omega t) \tag{3.9}$$

where $E(x,y)_t$ is the transverse dependence of electric field vector and $\beta$ is the modal propagation constant, which has discrete values for different modes. Maxwell's equations need to be decoupled in order to describe each field component separately. Equations can be decoupled by the curl operator $(\nabla\times)$ and the vector triple product is simplified by the operator identity

$$\nabla \times (\nabla\times) = \nabla(\nabla\cdot) - \nabla^{\mathbf{2}} \tag{3.10}$$

The decoupled equations then represent the homogenous vector wave equations, for the condition of no charged sources. Also fixing the coordinate system vector operator $\nabla^2$ can be replace by the scalar laplacian $\nabla^2$. With the above simplifications the wave equations for a dielectric waveguide structure leads to the homogenous wave equations.

$$\begin{aligned} \nabla^2 + n^2k^2\mathbf{E} &= -\nabla\mathbf{E}_t \cdot \nabla_t \ln n^2 && (3.11) \\ \nabla^2 + n^2k^2\mathbf{H} &= (\nabla \times \mathbf{H}) \times \nabla_t \ln n^2 && (3.12) \end{aligned}$$

Further simplification can be made to the homogeneous wave equation using weak guidance approximation. The weak guidance approximation considers that the electric are parameterised by the waveguide parameter V and the profile height parameter $\Delta$, these parameters are described in the next section. In weak guiding, i.e. $\text{n}_{co} \approx \text{n}_{cl}$, the $\Delta \ll 1$, for the case of small variation in the waveguide refractive index profile. To the zeroth order approximation the the Scalar Wave Equation is given by:

$$\nabla_t^2 + k^2n^2(x,y) - \beta^2\Psi = 0 \tag{3.13}$$

where $\nabla_t^2 = \frac{\partial^2\mathbf{\Psi}}{\partial x^2} + \frac{\partial^2\mathbf{\Psi}}{\partial y^2}$ for a 2 dimensional planar slab waveguide structure. For weak-guidance approximation TE and TM modes have the same propagation constant but transverse polarisations, in $y-$ and $x-$ directions respectively.

### 3.1.2 Scalar Wave equation for Planar Waveguide

The planar slab waveguide seen in figure 3.1 shows the core surrounded by and infinite cladding. The core half width equals $\rho$, and the refractive index is invariant in the $z$-direction or translationally invariant. Thus the refractive index is a function of $x$ and $y$ only i.e $n(x,y)$.

The boundary conditions for the scalar wave equations in a 2 dimensional planar waveguide are that

$$\Psi, \frac{\partial\mathbf{\Psi}}{\partial x}, \frac{\partial\mathbf{\Psi}}{\partial y} \tag{3.14}$$

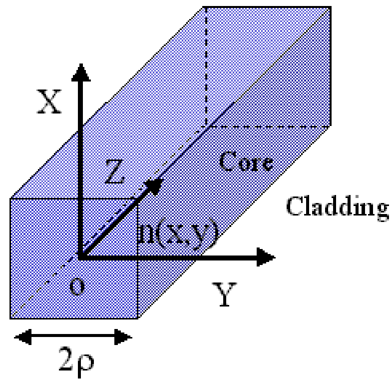



**Figure 3.1:** Refractive index of a planar slab waveguide is bounded by the core and cladding interface in the x and y directions, while being unbound in the z- direction. $\rho$ is the core half width and the refractive index ($n$) is translationally invariant, thus described only needs the transverse components, i.e $n(x, y)$.

are continuous across an interface between media of different refractive index, i.e. the core and cladding interface.

The modal field and eigenvalue equations are obtained by solving the scalar wave equation for bound modes in the core and cladding. Electromagnetic modal field decreases exponentially to zero in the cladding i.e. the cladding is large enough so that it can be taken as $\infty$. Symmetric or anti-symmetric mode fields are supported in the core. Here the waveguide parameters are introduced consisting of: core ($U$), cladding ($W$) and waveguide ($V$) parameters.

$$U = \rho(k^2 n_{co}^2 - \beta^2)^{1/2} \tag{3.15}$$

$$W = \rho(\beta^2 - k^2 n_{cl}^2)^{1/2} \tag{3.16}$$

$$V = 2\pi\rho/\lambda(n_{co}^2 - n_{cl}^2)^{1/2} \tag{3.17}$$

$$= k\rho\sqrt{2n_{co}\Delta_n} \tag{3.18}$$

$$\Delta_n = \frac{n_{co}^2 - n_{cl}^2}{2n_{co}^2} \tag{3.19}$$

$$V^2 = U^2 + W^2 \tag{3.20}$$

To verify the solutions are real in the core and cladding, a simple test can be made. The bound modes are substituted into equations (3.15) and (3.16). Values of $\beta$ correspond to normal or bound mode propagation constant. Thus were the bound modes of $\beta$ are in the range of values $kn_{co} > \beta > kn_{cl}$ the solutions are real. Solving the boundary conditions the eigenvalue equation is derived for odd and even propagating mode and are:

$$\Psi_{even} => W = U\tan(U) \tag{3.21}$$

$$\Psi_{odd} => W = -U\cot(U) \tag{3.22}$$

From the eigenvalue and $V^2$ equations (3.21, 3.22 and 3.17) give solutions discrete solutions for U, at fixed waveguide width and wavelength. It is then possible for the propagation constants to be found. They were found numerically using Maple and the appendix code section 8.2. Discrete eigenvalue solutions exist for the multiple modes and are dependant on wavelength, parameters of core geometry and the refractive index profile, an example of a solution is shown in Figure 3.2.

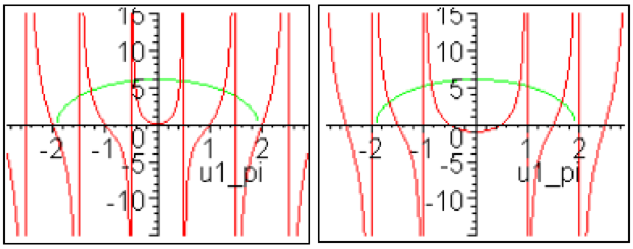



**Figure 3.2:** W is plotted from the eigenvalue equation (red) and $V^2$ equation (green) both in terms of U. The example has a constant $\rho$ (halfwidth) of 5$\mu$m and 1500nm wavelength. The even and odd mode solutions for bound modes is plotted, left and right respectively.

From the eigenvalue equation plots, possible solutions for even and odd modes fall in the following regions:

$$\begin{array}{ccccccc} 0 & < & \mathrm{U} & < & \pi/2 & & \Psi_0 \\ \pi/2 & < & \mathrm{U} & < & \pi & & \Psi_1 \\ \pi & < & \mathrm{U} & < & 3\pi/2 & & \Psi_2 \\ 3\pi/2 & < & \mathrm{U} & < & 2\pi & & \Psi_3 \end{array}$$

Note that when $V = U$ are mode cut off points, except for the fundamental mode, cut-off boundaries for planar waveguides are usually represented in terms of the $V$ number.

Numerical solutions can then provide the propagation constant for a specified waveguide width ($2p$) using $\beta^2 = k^2 n_{co}^2 - U^2/\rho^2$, thus using the effective index is can be determined using the planar wave approximation $n_e = \frac{\beta}{k}$.

### 3.1.3 Characterisation

In order for a design of the device to be efficient, acquiring a good understanding of modal effective index is required. Using the Maple program and its spread sheet package, data of effective indies as a function of waveguide width for particular refractive index profiles and wavelengths was collected, code is attached in appendix. In Figure 3.3 the modal effective indies for the multiple wavelengths 0.85, 1.3 and 1.55 $\mu$m are shown, were the refractive index profile is set to 1.5 and 1.527 for the cladding and core, respectively. It can be noted that each mode order has a particular symbol associated with it e.g. the square is the $\Psi_0$ mode or fundamental mode, and each wavelength has a designated colour e.g. blue is 0.85 $\mu$m.

The effective index for a particular wavelength is lower when the mode order is higher, and it also depends on the waveguide width. For a fixed mode and waveguide width the effective indies are roughly inversely proportional to wavelength, e.g 2$\mu$m for $\Psi_0$ mode in

Figure 3.3. For a large waveguide width (25$\mu$m), the effective index of all wavelength modes approach the core effective index (1.527). It is possible for two or more wavelengths to have similar effective index values as a function of waveguide width, if the longer wavelength propagates in a lower-order mode than the shorter wavelength. An example is where 850nm light propagates as $\Psi_2$ mode and 1550nm propagates as $\Psi_1$ mode. These modal effects become quite useful in designing the CWDM device proposed in this thesis and will be used in the modal manipulation section 3.3.

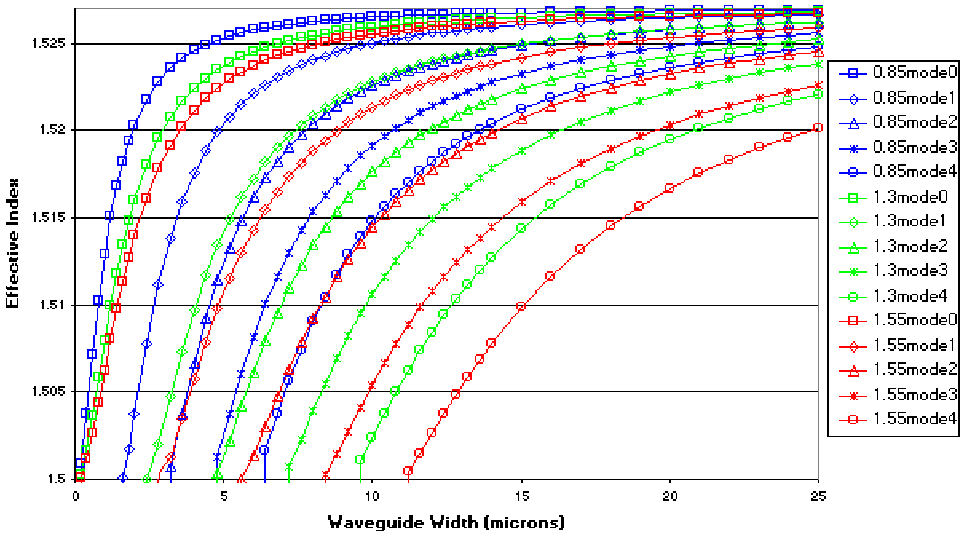



**Figure 3.3:** Modal effective index characteristic for the first five bound modes of the step-profile slab waveguide, as a function of waveguide width. The telecommunication wavelengths (850, 1300 and 1550nm) are shown, where the refractive indies are is 1.5 and 1.527 for the cladding and core, respectively.

Understanding of the modal effective index for different step refractive index profiles was also investigated. Figure 3.4 considers the $\Psi_1$ mode at a constant wavelength for several refractive index difference cases. Increase of the refractive index of the core increases the rate of change of the effective index of propagating modes as a function of waveguide width; also the cut-off width is shifted to smaller dimensions. Wider waveguide gives higher effective index values, however this is limited to the refractive index of the core for bound modes.

An alternative method of observing the effective index matches is shown in Figure 3.5, in this plot higher order modes are plotted as a function of (s-p) where s is a reference width and p is the waveguide width variable. The advantage of this method is that the data is compressed and it indicates the waveguide widths that correspond to the fundamental mode effective index matching to a higher-order mode at each cross-over point. This is important as the input and output of the CWDM device needs to be in the fundamental mode. Disadvantage of the cross-over method is that it focuses on waveguide widths that represent only a small portion of possible effective index matches between propagating super-modes per wavelength.

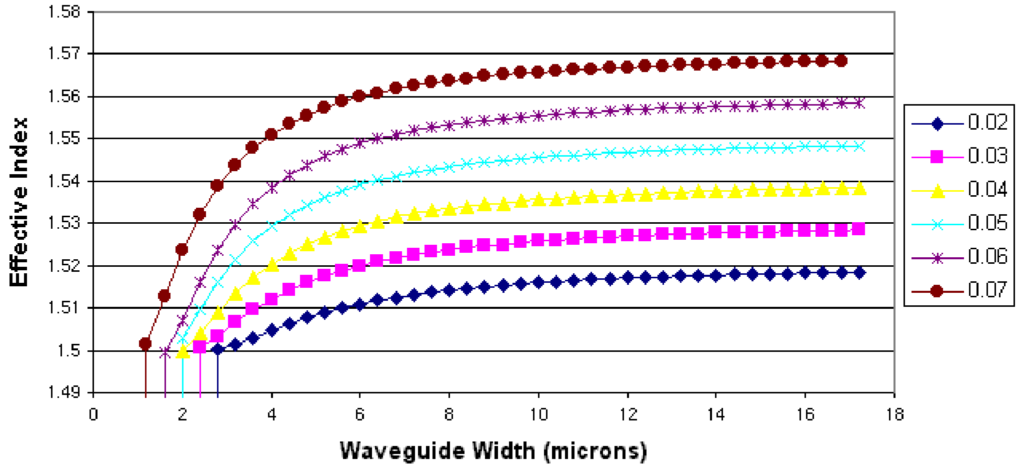



**Figure 3.4:** The modal effective index characteristic for changes to the refractive index difference of the core and cladding (0.02-0.07). The parameters fixed are the $\Psi_1$ propagating mode, the operating wavelength (1300nm) and the cladding refractive index of 1.5.

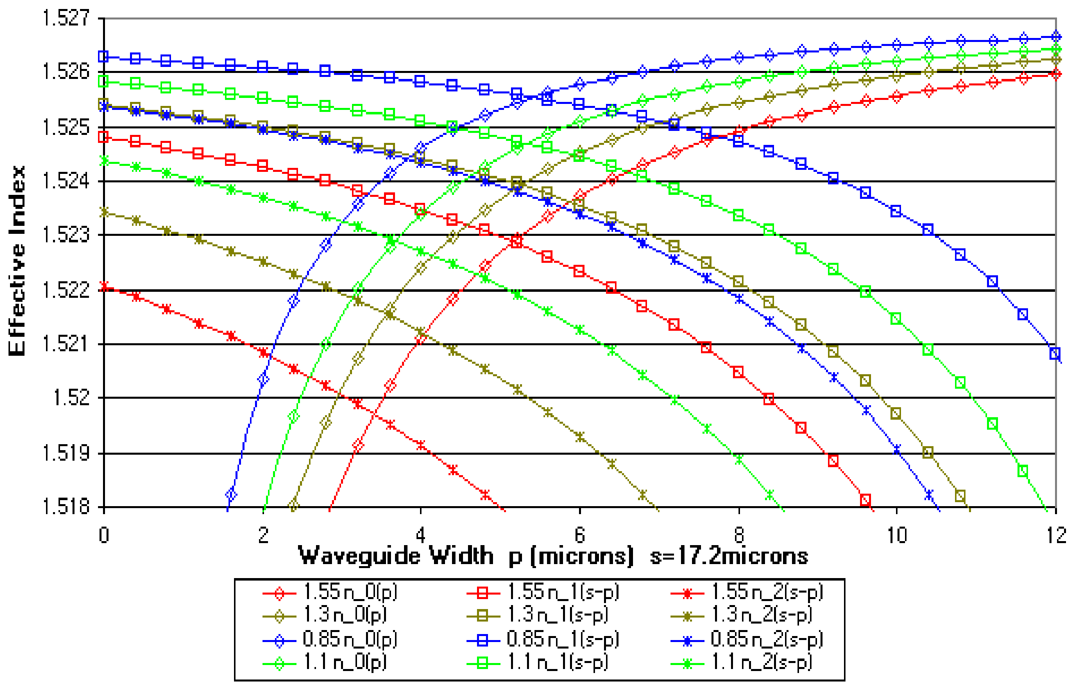



**Figure 3.5**: Three modal crossover plot.

## 3.2 Super-Modes

Super-modes are modes that describe the field properties across the entire cross-section of the waveguide structure. For multi waveguide structures one may think of the super-mode as a superposition of waveguide modes. Symmetric Y- junctions are used here to demonstrate super-mode behaviour (theory of super-mode in Y-junctions seen in [20]).

The symmetric Y-junction has one wide arm (bigwidth arm) and two equal width arms. For simplicity a sum of the two other arms is equal to the bigwidth arm. The two arms need to be separated slowly enough for mode coupling to be avoided. An indication of appropriate waveguide separation, shown in Figure 3.6, for relative to the refractive index difference in core and cladding of waveguides. Approximately 12$\mu$m separation is required for the polymer based material used in this thisis. Modal evolution is examined by simulating the Y-junction in the forward and reverse input directions, i.e. one possible input arm in the forward direction and one or two input arms in the reverse direction.

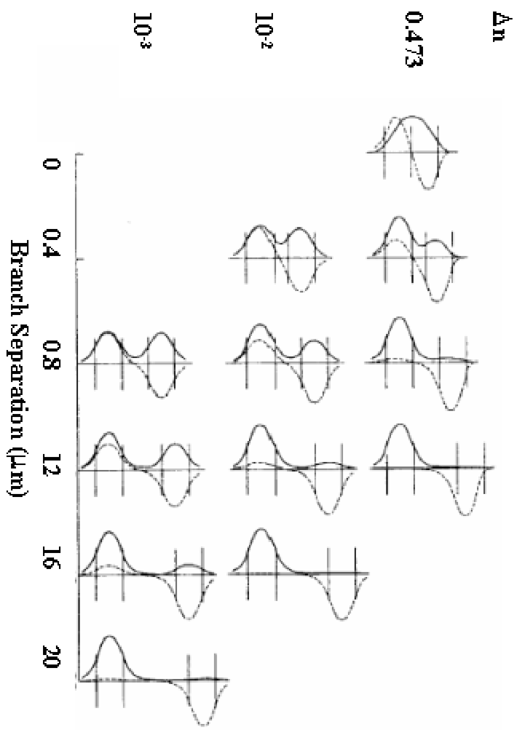
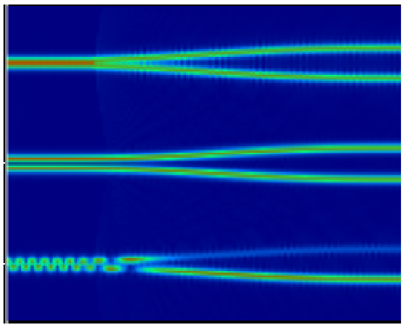


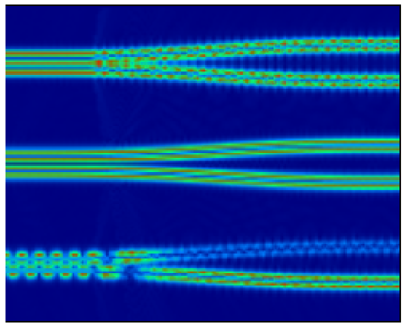

**Figure 3.6:** Supermodes of a coupled waveguide system shown with varying degrees of guide asymmetry, $\Delta n$ and guide separation [20].

### 3.2.1 Forward Direction

Simulation of super-modes in the forward Y-junction is investigated, where the bigwidth is 6 and 12$\mu$m. In Figure 3.7(a) the top Y-junction is initially excited by the $\Psi_0$ mode, this is a symmetric mode. By symmetry of the Y junction the $\Psi_0$ evolves into the symmetric super-mode where equal power is split into the two $\Psi_0$ modes, one in each output arm,



(a) Bigwidth $6\mu$m, top to bottom simulations inputs are $\Psi_0$,$\Psi_1$ and then both $\Psi_0$ and $\Psi_1$



(b) Bigwidth $12\mu$m, top to bottom simulations inputs are $\Psi_2$,$\Psi_3$ and then both $\Psi_2$ and $\Psi_3$

**Figure 3.7**: The field evolution in the forward direction of a Y-junction

and with equal phase. The anti-symmetric mode is not involved in the above case, but if the $\Psi_1$ mode is excited in the input the anti-symmetric super-mode will be evolved into the outputs, where each output arm propagates in the $\Psi_0$ mode with equal power but opposite phase.

Exciting the input with both the $\Psi_0$ and $\Psi_1$ modes of equal power it can be observed in the simulation at the bottom of Figure 3.7(a) that only one output arm propagates in the $\Psi_0$ mode. This is a result of the interference of the symmetric and anti-symmetric super-modes. In one arm both $\Psi_0$ modes of the two super-modes are 'in-phase' but in the other arm they are out-of-phase by $\pi$ causing destructive interference. To control the output power the phase of the odd mode needs to be changed, a $\pi/2$ change will cause approximately equal output power, while an 'in phase' or $\pi$ out of phase cases will switch approximately all the power into one or the other output arm.

Similarly in Figure 3.7(b) combining $\Psi_2$ and $\Psi_3$ modes at the input, $\Psi_1$ can become the output of only one arm. In this case the mode transformation in taper is a little more complicated then in the previous example. To avoid radiation, the waveguide width of the input is assumed to support the $\Psi_3$ mode, and the two output arms support the $\Psi_1$ mode. Due to the taper transition there exists an uneven power distribution in the $\Psi_1$ output mode in both arms. This is generated by the $\Psi_2$ input mode. The $\Psi_3$ input mode also evolves the $\Psi_1$ mode into each output arm. The difference in the two evolved super-modes is that the symmetric super-mode is generated by the $\Psi_3$ input mode, while the anti-symmetric super-mode is generated by the $\Psi_2$ input mode. As the overlap of $\Psi_1$ output modes are not exactly 1, totally destructively interfered in one output arm is prevented. Therefore the interference of one arm will not be as effective as in the previous case, but will still occur due to interference.

### 3.2.2 Reverse Direction

Investigated here are the two cases with 6 and $12\mu$m bigwidths but in the reverse direction, also a bigwidth of $2.5\mu$m shows some interesting results. The reciprocity law for lossless devices should be satisfied with the Y-junction operating in the opposite direction.

Consider only one of the two inputs being excited by the $\Psi_0$ mode, and compare the bottom Y-junctions in Figures 3.7(a) and 3.8(a). The reverse results evolve into a beating field, and comparing the with forward case, this resembles the $\Psi_0$ and $\Psi_1$ modes in the bigwidth, thus both the anti-symmetric and symmetric super-modes are excited at the input and reciprocity is restored.

Unlike the forward-propagating Y-junction the reverse propagation along the two input modes can have different phase. Shown are the 'in-phase', $\pi/2$, and $\pi$ out-of-phase cases. The 'in-phase' case corresponds to the forward propagating device. The $\pi/2$ out-of-phase resembles a single input case, where both inputs excite the two super-modes and in the tapering section the super-modes add to give a higher amplitude super-mode with a phase offset of $\pi/4$. The $\pi$ phase change in two input modes results in the symmetric super-mode destructively interfering, while the anti-symmetric mode adds constructively, thus the $\Psi_1$ propagates in the single output. These results are reiterated in comparing Figures 3.7(b) and 3.8(c), but considering the higher-order input mode $\Psi_2$ and also the bigwidth is increased to $12\mu$m such that the higher-order modes are supported.

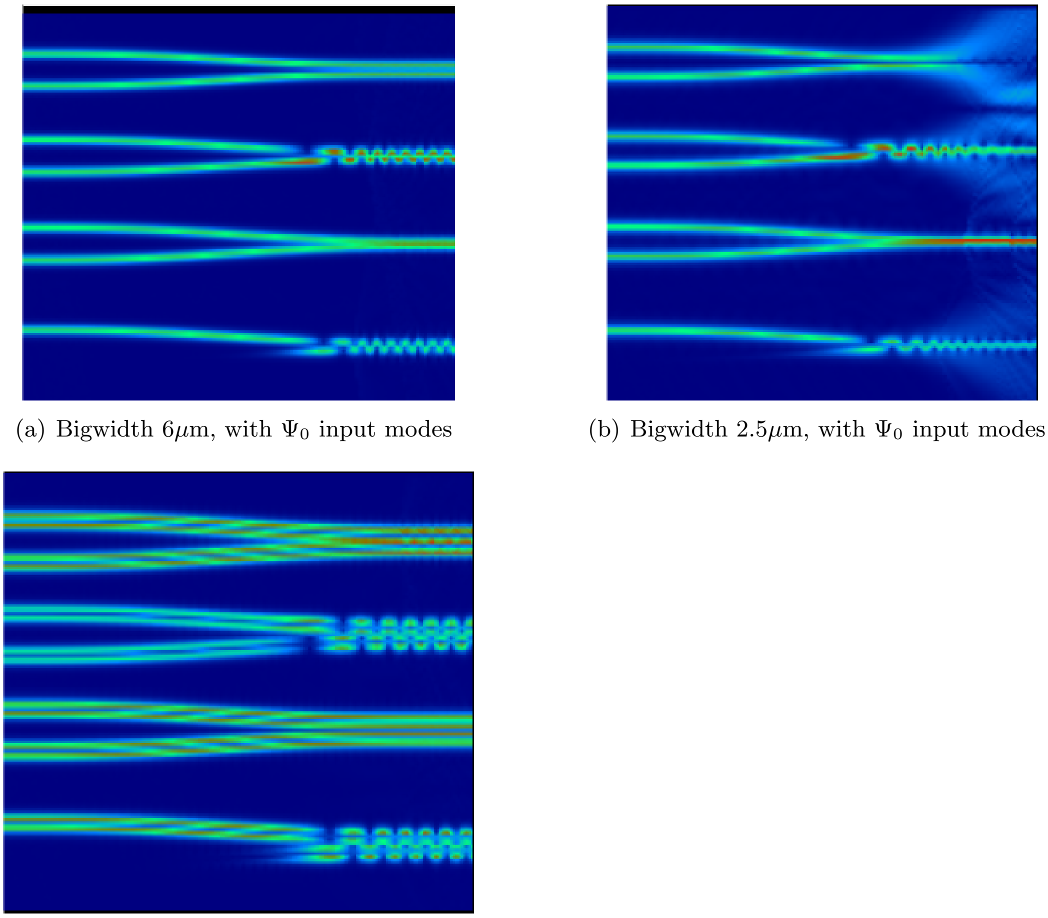
What happens if the $\Psi_1$ modes are not supported or are cut off in the single output arm case? The principles described above still hold but the anti-symmetric mode becomes a radiative mode and converts the mode power into radiation. Shown is in Figure 3.8(b) are the same input conditions as the $6\mu$m bigwidth case except the bigwidth is set to $2.5\mu$m. Thus in the single input and $\pi/2$ phase offset cases half the input power to radiated, while for $\pi$ phase offset case all power is radiated. The 'in phase' case does not loose power as the anti-symmetric mode is not excited.

### 3.2.3 Asymmetric Waveguide Junctions

As seen earlier the adiabatic Y-junction device, Figure 2.2(c), was used to filter wavelengths into two different output arms. The same principles discussed in the symmetric Y-junction apply to asymmetric Y-junctions. The exception is that each mode and wavelength have more options in choosing the best super-mode to evolve into. For example the Y-junction devices evolved into either the fundamental or $\Psi_1$ mode dependant on wavelengths and the output waveguide widths. To be considered in this thesis are 2-, 3-, 4-, and 5-way asymmetric junctions, these are studied in the adiabatic bend length section 4.1. Were the 2- and 3- wavelength demultiplexing is explored with the aid of dummy output arms.

## 3.3 Modal Manipulation Methods

What are the best waveguide structures to create 2 and 3 wavelength division multiplexing/demultiplexing devices, given the effective index is wavelength and waveguide width dependant? In an ideal situation, each wavelength's effective index at input and output of a junction is matched such that adiabatic evolution through the junction is loss-less, this is difficult to achieve in reality when considering multiple wavelengths. The process of



(a) Bigwidth 6$\mu$m, with $\Psi_0$ input modes

(b) Bigwidth 2.5$\mu$m, with $\Psi_0$ input modes

(c) Bigwidth 12$\mu$m, with $\Psi_1$ input modes

**Figure 3.8:** The field evolution in Y-junction in the reverse direction. Phase offset between inputs for the top three Y-junction are $\pi, \pi/2$ and 0 top to bottom, finally a single input arm is excited in the bottom simulation.

designing the device begins with opimising the Input-Stem junction and then continuing to optimise the Stem-Output junction, the junction may be a Y or n-way.

### 3.3.1 Design Obstacles

#### Wavelength Selection

Wavelengths selected initially consisted of 1300, 1425 and 1550nm but the separation was found to be too small for demonstrating an effective 3 channel CWDM device with a single output junction. Thus the separation was increased to fill the whole telecommunication spectrum with 850, 1300, 1550nm. Also the change in modal effective index as a function of wavelength was not significant enough when only one particular higher order stem mode was evolved into for the telecommunication spectrum.

Evolution can be optimised for one wavelength with high efficiency. But optimising for multiple wavelengths the efficiency of evolution at the Input-Stem junction becomes poor, due to compromises in the input and stem widths to give the best match of effective index for all operating wavelengths.

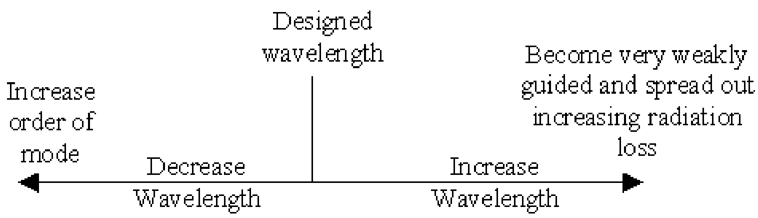



**Figure 3.9**: Design issues when considering multiple wavelengths.

Figure 3.9 gives the basic characteristic of a constant width waveguide as a function of wavelength. For CWDM the wavelength separation is large and therefore optimising for multiple wavelengths the consideration of the following effects are required, such as increasing the number of possible supported propagating modes when wavelength is small and greater loss to radiation as higher wavelength become cutoff.

### Stem Width Conditions

The stem waveguide width restricted by the adiabatic operation of the device requiring slowly varying waveguide structures. The condition for the stem width are:

i $\sum$ inputs arm widths = input stem; and

ii Output stem width = $\sum$ outputs arm widths

Initially the condition of the input and output stem widths remained equal. Problems relating to this condition included the output stem width that would be too small if the output arm modes were to consist of more than one higher-order output mode. Why is this a problem? The device operates adiabatically and width miss match prevents adiabatic evolution. This is shown in Figure 3.10, by removing the phantom arm of the 3 wavelength demultiplexing device that was shown in Figure 2.3. In the absence of the input phantom arm only a small percentage of the $\Psi_1$ mode is excited in the stem due to mode coupling, that then evolves into the top narrow output arm as designed. The output of the device is significantly changed, with the majority of input power exiting the bottom output arm, this arm previously had no output!

The stem width condition was later relaxed and an immediate degree of flexibility in device parameters was utilised. The two component stem consisted of two stem widths connected by an adiabatic taper. The taper has the disadvantage of increasing the device length. But the advantage is the independent optimisation of the input-stem and stem-output junctions.

### Input-Stem Transition Trends

Simulations of Input-Stem junction evolution was tested with first the stem width fixed while varying the input width, and then visa versa. When the input width is small the

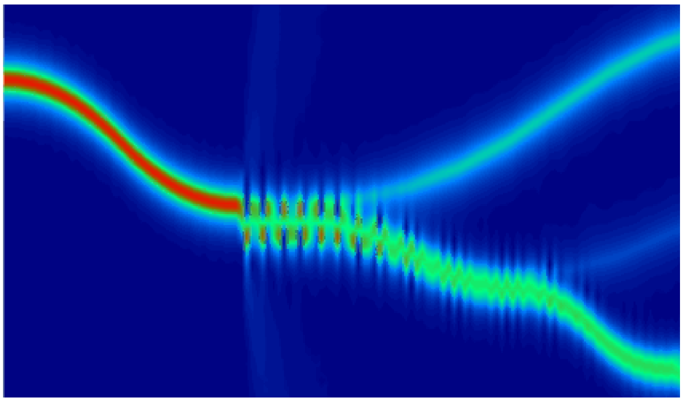
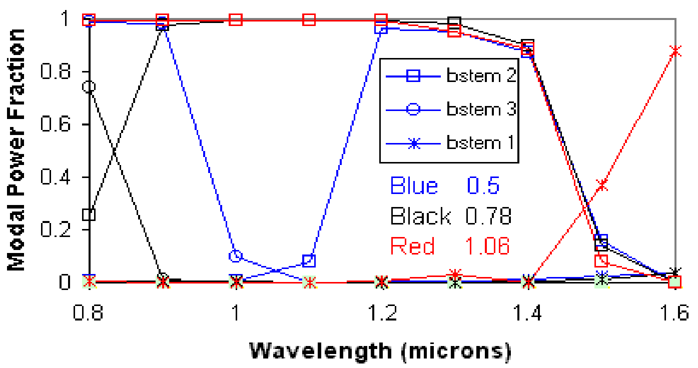


**Figure 3.10:** Input and stem width miss match is shown in an extreme case, the phantom input arm is totally removed from the previously simulated concatenated Y-junction demultiplexer operating at 1550nm.

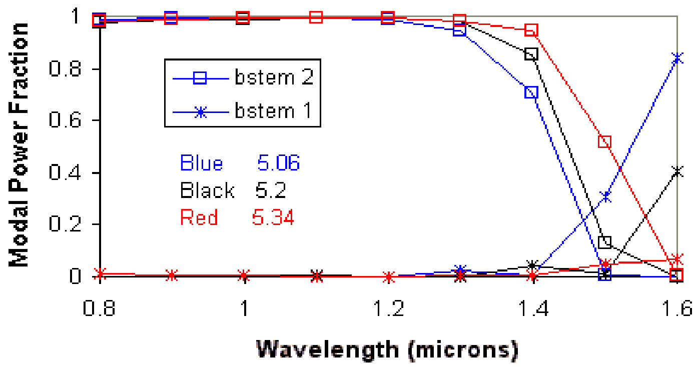
highest-order mode that is supported is evolved into. For slightly cut off modes the small input width couples into a radiative mode, thus that particular mode/wavelength is highly attenuated. Increasing the input width allows the effective index of the input super-mode to become closer matched to lower order stem modes. Thus it can be observed that the modal power fraction in the evolved stem modes begins to favour lower-order modes as the input width is increase, while the higher-order mode fraction decreases until finally the lower order mode remains.

If the input width is fixed and the stem width is decreased it also can be observed that this trend of effective index match of super modes still occurs and is consistent. The crossover point reflects that another mode has a closer effective index match. Increasing and decreasing the input or stem width respectively will shift the cross over point to lower wavelengths. The magnitude of the shift is about 2 times greater when the input width is changed instead of the stem width.

An example is provided in Figure 3.11. Input and stem widths will support and evolve different modes with the change in wavelength. Simulation of the input-stem transition effects have been observed by using 850nm, 1300nm and 1550nm light sources and keeping the stem fixed to 5.2$\mu$m while varying the input width. Effective evolution of 1550nm light does not occur for the $\Psi_2$ mode with a maximum of 0.1 modal overlap fraction. The $\Psi_1$ mode is evolved into when the input width is greater then0.9$\mu$m. An input width equal to 0.5$\mu$m allows the evolution of 850 nm into the $\Psi_3$ mode and 1300nm into $\Psi_2$ mode. At an input width of 1$\mu$m the 850nm $\Psi_3$ mode is cut off and the evolution into mode $\Psi_2$ begins to dominate, while 1300nm light begins to be more effectively evolved into mode $\Psi_1$.




(a) Varying the Input Width (0.5, 0.78, and 1.06$\mu$m) with stem fixed to 5.2$\mu$m




(b) Varying the Stem Width (5.06, 5.2 and 5.34$\mu$m)with input fixed to 1$\mu$m

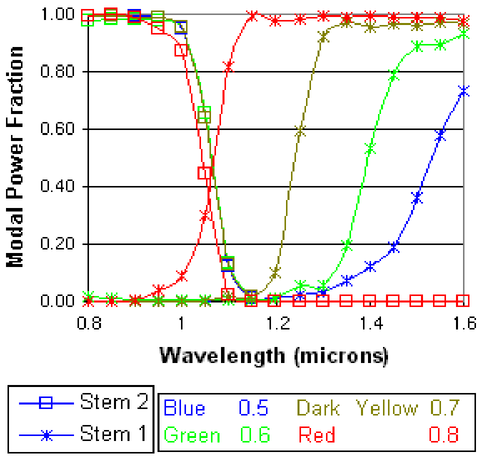
**Figure 3.11**: Input-Stem junction tuning trend

### Input-Stem Cutoff Effect

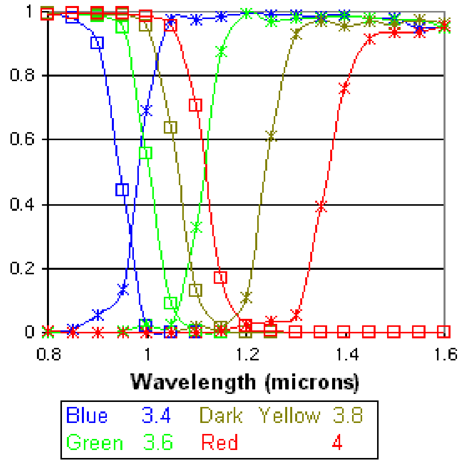
Cutoff effect focuses on the evolution of multiple wavelengths into the stem but in different stem modes. The idea is to select a stem and input width combination that has the following properties

- Longer wavelength (1550nm) higher-order stem modes are cut off while the shorter wavelength (1300, and 850nm) higher-order stem mode can still propagate.
- The input width is small such that the longer wavelength evolves into the low order mode while the shorter wavelength evolves in the higher-order mode.
- The input width must be large enough to reduce the longer-wavelength coupling into radiative modes.

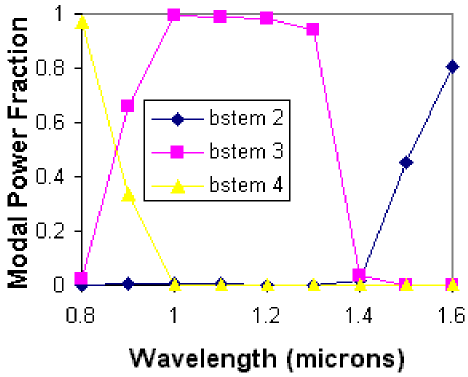
The cutoff stem widths for 1550nm at $\Psi_1$, $\Psi_2$,and $\Psi_3$ modes were investigated, shown in Figure 3.12. For example, a 5.2$\mu$m stem width corresponds to approximately 1.512-1.517 effective index values for $\Psi_3$ mode for 850nm and $\Psi_2$ mode at wavelengths between 1300nm and 1550nm. $\Psi_3$ mode is just cut off for 1550nm, i.e. the mode is not supported. Wavelength separation into different stem modes was achieved. It was found that the higher the mode order that is cutoff, the more modes that can be coupled into over the 800-1600nm spectrum. Smaller inputs enhanced the cutoff wavelengths effect. Advancing to the cutoff width for mode $\Psi_3$ at 1550nm wavelength it was found that in the 800nm to 1600nm range modes 2, 3 and 4 could be evolved into with high efficiency of 90+%.



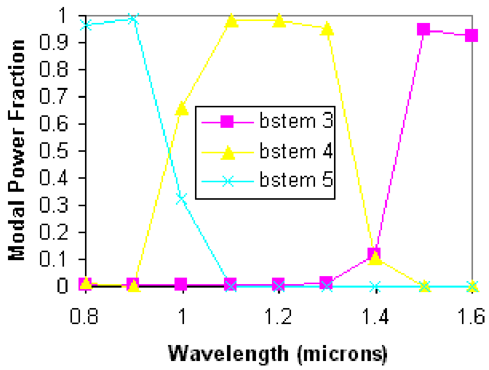

(a) Varying input width (0.5-0.8$\mu$m) for a fixed stem width of 5.2$\mu$m. The stem width corresponding to the cutoff of $\Psi_2$ mode at 1550nm. It is observed that higher wavelengths propagate in the $\Psi_1$ mode and lower wavelengths propagate in $\Psi_2$ mode. The $\Psi_1$ mode is mainly affected by changing the input width,




(b) The stem width is varied (3.4-4$\mu$m) with a fixed 0.75$\mu$m input width. Changing the stem width shifts both propagating stem modes, with the cross over of modal power fraction being reduce with increased stem width.




(c) mode3 cutoff




(d) Mode4 cutoff

**Figure 3.12**: Spectra of input-stem mode evolution using slightly cut of stem widths.

#### Stem-Output Tuning

The Stem-Output junction ideally requires the stem-modes to match super mode effective indexes that create a fundamental propagating mode in one output arm per channel. The effective index characteristic of the stem is determined by the width, wavelength and propagating mode. Assuming approximate effective index match, the fundamental output mode can be evolved into an arm with width that corresponds to the stem-mode. This is approximate as the output modal super-mode will have a slightly lower effective index then the effective index of the isolated output arm width. For a case where all the wavelengths are in the same stem mode the fundamental mode waveguide width was found to be very similar, this is not good for manufacturing tolerance.

If the stem modes are different it is possible for the modes to have approximately equal effective index values, this tending to results at large stem widths. Then repeating the effective index match between stem modes and the fundamental modes the output waveguide widths can be approximated. With different channels in consecutive stem modes, where the smaller wavelength channel are higher order modes, a much greater difference in output arms can be achieved with respect to the single mode case. The disadvantage of this method is that the output arms for one channel may also match other channel higher-order modes which would increase crosstalk.

#### Bi-directional Operation

Demultiplexing and multiplexing operation in a bi-directional device needs to be similar. The device is initially designed in the demultiplexing mode due to the greater difficulty of spectral filtering then spectral combining. It was discovered that larger input stem width devices failed to multiplex. This is due to the phantom arm approximating the stem width and thus propagating modes do not have the closest match of effective index with the desired multiplexing output arm, causing the channels to continue propagating in the phantom arm. This can be fixed by adding additional phantom arms.

Chapter 4

# Adiabatic Evolution of Modes

## 4.1 Adiabatic Bend Length

Adiabatic evolution is a fundamental mechanism employed by the proposed CWDM devices in this thesis. The question of how gradually an input/output arm has to combine or diverge in order to create an adiabatic bend is considered in this section. Bent arms are defined using sinusoidal bends, where the bend radius is determined as a function of the offset. The function for the middle line of the sine bend for a given bend length ($L$) is $x = s \times L, y = 0.5 \times \sin(2\pi s)/\pi$), with $s$ varying from 0 to 1.

At small bend lengths the output modes are expected to resemble mode coupling, due to a sudden translational change in the waveguide structure. Adiabatic bend length is the length that approximately constant modal output powers are recorded with further increase to the bend length. Also for practical applications in devices, it is required that there is minimal crosstalk, insertion loss and excess loss. Therefore the evolved power fraction should be high in the designed output arm and minimal power evolved into other arms. Adiabatic bend length is expected to be realised for longer bend lengths. To determine the adiabatic bend lengths, simulations of 3, 4 and 5 way junctions are investigated.

### 4.1.1 Adiabatic Bend Length for 3-Way Asymmetric Junction

The simulation for 3-way asymmetric junctions was undertaken using 4 duplicate junctions, each excited by a different mode $\Psi_1$, $\Psi_2$, $\Psi_3$ and $\Psi_4$ from bottom to top simulation respectively, as shown in Figure 4.1(a). These modes are then propagated through the 3-way junction. An asymmetric junction here consists of non-equal output arms that sum to equal the input arm width.

It was found in Figure 4.1(b) that higher-ordered modes required smaller bend lengths for an adiabatic transition. The adiabatic bend length was found to be 7mm for the tested modes, as shown in Figure 4.1(b). The higher order modes ($\Psi_{2,3,4}$) excited multiple modes for bend lengths less then 2mm. Mode swithcing is depicted in Figure 4.1(a), this is driven by the effective index match of the super-modes. As each of the modes have different effective index values, the fundamental having the highest value, the output are all unique.

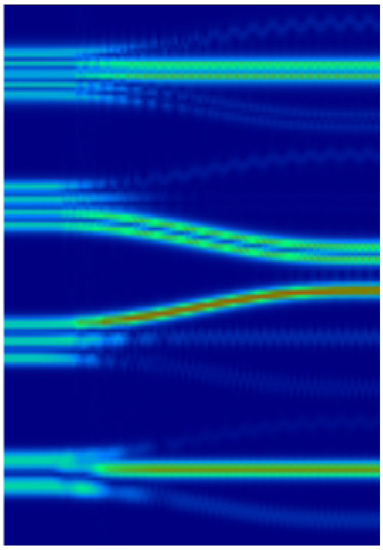
### 4.1.2 Adiabatic Bend Length for 3-Way Symmetric Junction

To simplify the adiabatic bend length analysis a 3-way symmetric junction was studied, i.e. the two outer arms having equal widths and geometry. The above method for the asymmetric junctions was reused. In the symmetric junction case the expected results for the output modal power fraction was: the two even arms would have equal power of approximately 50% or the middle output arm would have the closest effective index to the input mode and thus evolve approximately 100% of the input power. The middle arm was made smaller than the two equal arms, allowing modal evolution to be driven into the outer arms due to the close effective index values.

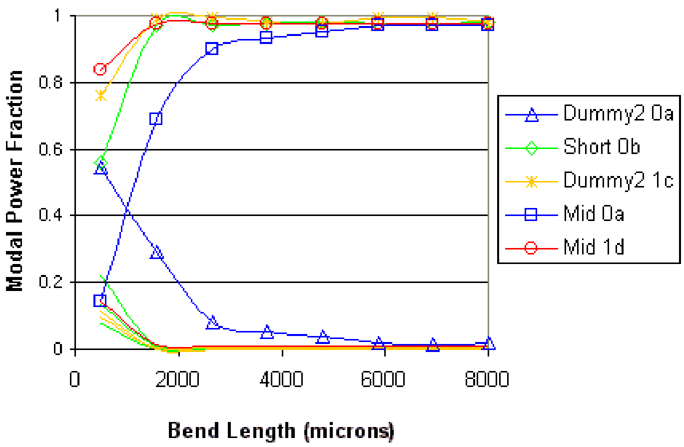
Adiabatic bend length, in Figure 4.2(b), was found to be significantly reduced from the asymmetric bendlength case, to 2.5mm although the modes that coupled into the two outer arms showed mode coupling characteristics, the greatest mode coupling occurring in the lower order modes. This reflects that lower order modes need to be least deviated []. Below 2.5mm bend length multiple mode are excited in the junction. Beating in the $\Psi_2$ mode case is small but is affected by both outer arms.

Does the 50% division of power hold for the telecommunication wavelength spectrum, and for the purpose of a realistic device in the presence of a mixture of propagating stem modes? To simulate this an asymmetric Y junction becomes the input-stem junction and is connected to the stem-symmetric junction, then the simulation is swept over the telecommunication spectrum of 800 to 1600nm. Relatively high fundamental power in the middle arm and approximately equal power distribution in the two even output arms could be achieved in the 5, 2 and 5$\mu$m test case over the whole spectrum. But in the 3.2, 5.6 and 3.2$\mu$m output test case the mixing of modes caused large power difference in the two even output arms.

If multiple super-modes are evolved into, this occurs if there is a large miss match in effective indies of bound modes. The multiple supermodes interfere and create a beating effect in the output arms. The beating in the junction for non-pure modes is exemplified in Figure 4.3(b), for the 1100 to 1300nm wavelength range.

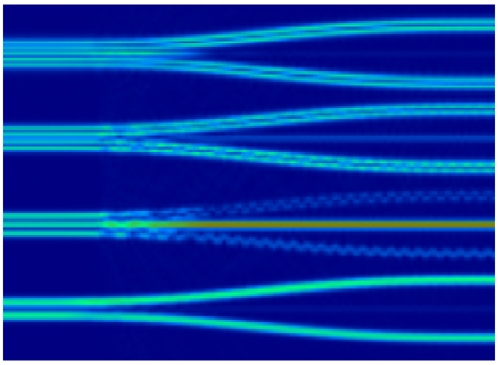


(a) Field plot at the bend length equal 6.9mm

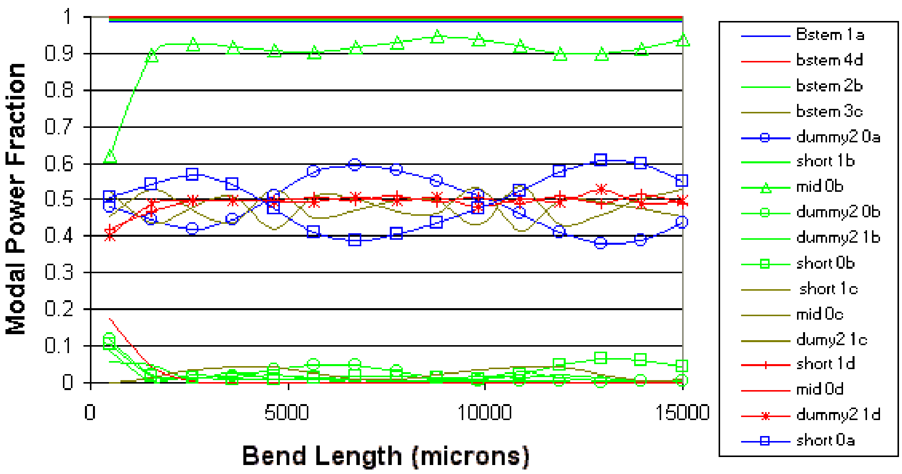



(b) Asymmetric 3, 4, 5$\mu$m junction bend length characteristic. In the legend a, b, c, and d refer to the input modes $\Psi_1, \Psi_2, \Psi_4$ and $\Psi_4$ respectively.

**Figure 4.1:** Asymmetric bends tested as a function of bend length for modes $\Psi_1, \Psi_2, \Psi_4$ and $\Psi_4$. Aysmmetric junction output arms are 3, 4 and 5 $\mu$m and the centre of output arms are offset by 9$\mu$m.

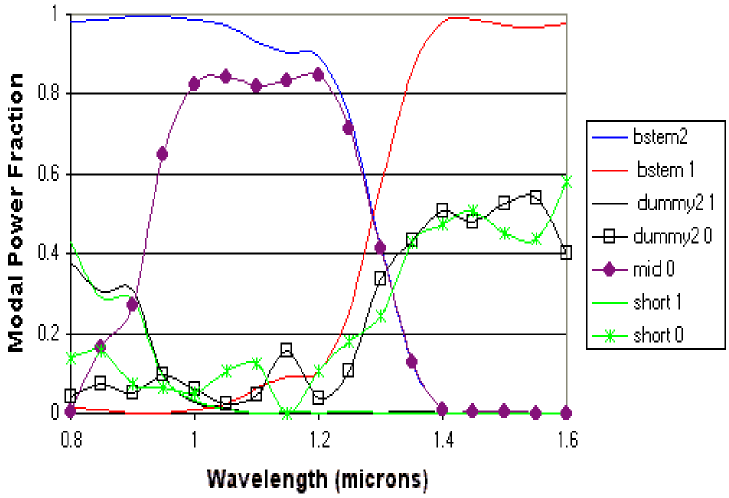


(a) Field Plot where bend length is 4.5mm and the two output arm centres separated by 12$\mu$m.

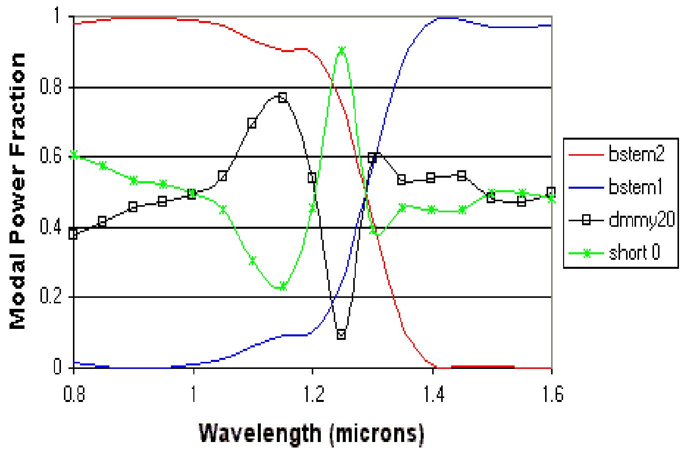



(b) Bend Length characteristic for evolution of pure modes. In the legend a, b, c, and d refer to the input modes $\Psi_1, \Psi_2, \Psi_4$ and $\Psi_4$ respectively.

**Figure 4.2:** Model Symmetric bends tested for bend adiabatic length for modes $\Psi_1, \Psi_2, \Psi_4$ and $\Psi_4$.




(a) 5, 2, 5$\mu$m output arm widths



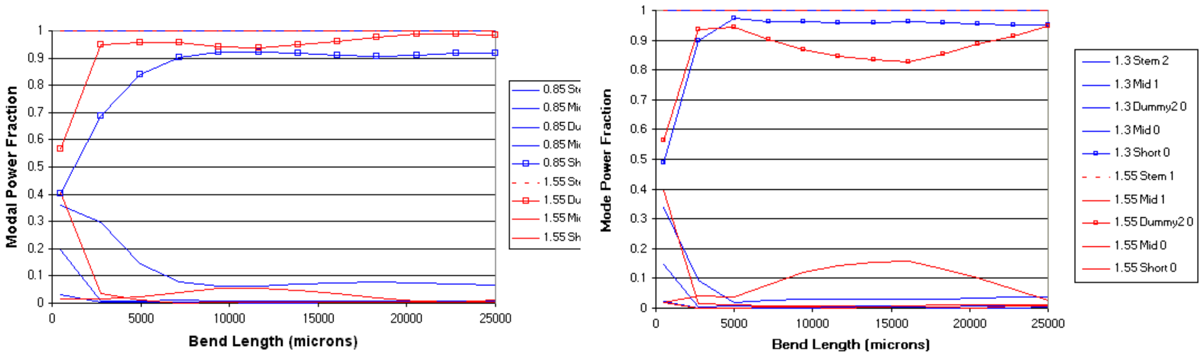

(b) 3.2, 5.6, 3.2$\mu$m output arm widths

**Figure 4.3:** 3-way symmetric junction with large bend length. Non-pure stem modes lead to uneven power distribution between equal arms.

### 4.1.3 Proposed Device Output Adiabatic Bend Lengths

Adiabatic bend length for output junction is dependant on propagating input mode, structure of junction, wavelength, waveguide separation and refractive index. Thus specific asymmetric junction structures are investigated for optimal bend lengths. The effective indices from the Stem-Output junction arms are closely matched. The parameters selected are used in the 2 and 3 wavelength division demultiplex and multiplexing devices that are described in chapter 5.

#### 2 Wavelength Devices

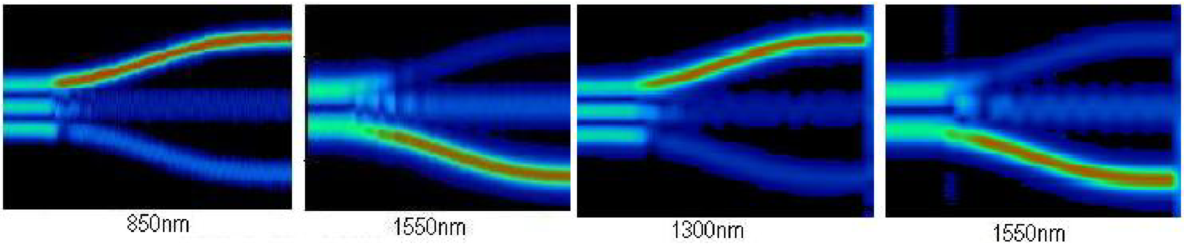



(a) Type 1 (b) Type 2

**Figure 4.4:** Adiabatic bend length determination for *Type*1 850nm and 1550nm and *Type*2 1300nm and 1550nm demultiplexing 3way junctions. Plot shows the output modal fractions as a function of bend length

A simulation of the 3-way stem-output junction transition for the 2 types of two-wavelength demultiplexers are shown in Figure 4.4. The demultiplexer type 2 required a longer bend length, primarily due to 850nm wavelength. Generally bend lengths less then 3mm couple significant amounts of modal power into multiple output modes and arms or constituent output arm local modes. Mode coupling is not desired as it is potentially a large source of crosstalk in a demultiplexing device, thus longer bend lengths are required. Longer wavelengths appear to have a small amount of beating between the output arms. While shorter wavelengths require a larger bend length for the adiabatic evolution to be realised.

The field plots of the optimised bend lengths are depicted in Figure 4.5, where the 3-way junctions demultiplexer type 1 and 2 optimum bend lengths are 9.4mm and 5mm respectively. These plots assume that only pure modes propagate at the input of the junction for each designed wavelength. Also the switching ability is highlighted with the two wavelengths going to the top or bottom output arms with minimal power leakage into the other two arms.



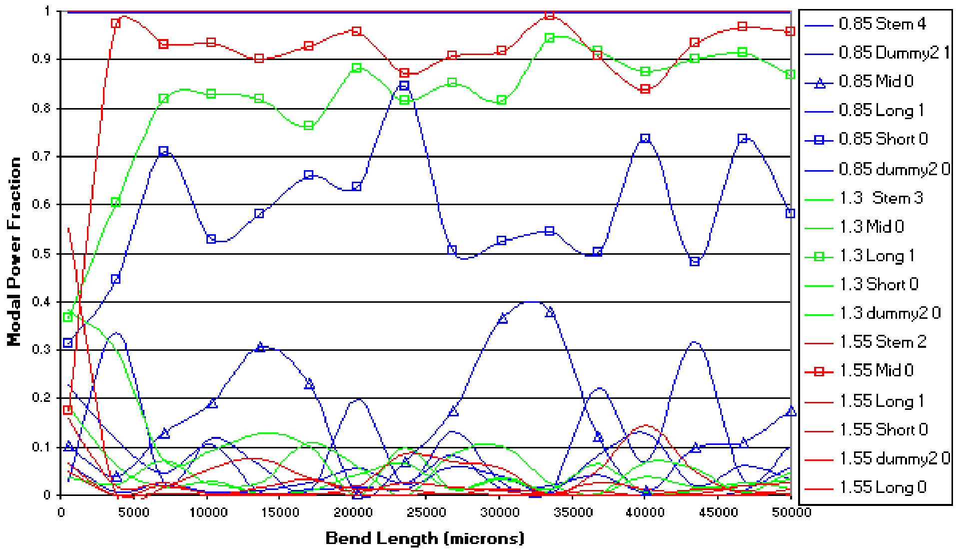

(a) Type 1 bend length of 9.4mm (b) Type 2 bend length of 5mm

**Figure 4.5**: Simulated field for both 2 wavelength CWDM at the optimised bend lengths.

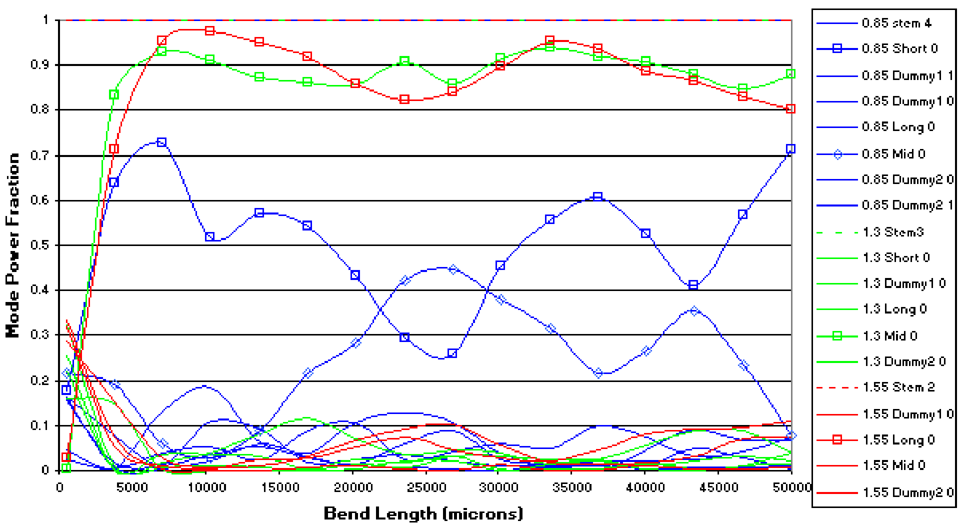
### 3 Wavelength Devices

Investigation on the effect to adiabatic bend length with increasing the number of output arms, as up to 4 or 5 output arms are used in proposed 3 wavelength demultiplexers.

The 4-way junction properties in Figure 4.6(a) evolve 0.8 and 1.55$\mu$m into $\Psi_0$ output mode while 1.3$\mu$m is evolved into the $\Psi_1$ output mode into separate output arms. Adiabatic bend length is again investigated: 1.55 and 1.3$\mu$m were found to have short adiabatic lengths of about 7-10mm, with about 90% and 80% power throughput respectively. 0.8$\mu$ is generally not as efficient due to mode coupling between the adjacent short and mid arms both with similar waveguide widths. The max peak of 85% at 22.5mm is due to this mode coupling effect. A peak of 70% is achieved at 7mm, this providing a more practical length device.




(a) 4-way junction




(b) 5-way junction

**Figure 4.6:** 4 and 5 output modal fraction as a function of bend length for, junction parameter to be used in 3 wavelength CWDM devices operating at 850 nm, 1300 nm, and 1550 nm wavelengths.

The 5-way device separates all 3 wavelengths above into the fundamental mode. It can be observed that the power throughput for 1.3$\mu$m is improved at the shorter bend length region. 1.55$\mu$m requires a few more millimetres to become adiabatic and due to the reduced width of the neighbouring waveguide a small amount of mode coupling at longer bend lengths is noted. The mode coupling is significantly changed in the 0.85$\mu$m wavelength, such that 30% throughput now corresponds to the 4-way maximum. The 0.85$\mu$m shorter bend length is slightly improved, but is still poor.

Adiabatic bend lengths for these selected cases are unexpected, as it was thought that the adiabatic bend length would increase significantly with an increase in the number of output arms. Observing the 1550 nm wavelength, there is a small increase in the shortest possible adiabatic bend length as the number of inputs is increased. The bend length increase is 1mm and 3mm, respectively, in going from 3 to 4 and from 4 to 5 outputs . However, as with the 2-wavelength case, it is the shortest propagating wavelength that is the limiting factor for adiabatic bend lengths.

### 4.1.4 Isolation of Operational Output Arms

An attempt to minimise mode coupling issues was attacked by separating the operational output arms is shown in Figure 4.7. Improvement is seen in the short length efficiency of the short wavelength 850nm from 70% to almost 80% , but this requires a doubling of the bend length to 12mm for optimum throughput for all wavelengths. The cost is that the longer wavelengths of 1300 and 1550nm have a 10-15% of throughput. Also the mode coupling seems to be enhanced for all wavelengths. The mode coupling is focused to the two nearest neighbours for the short and long wavelengths. The attempt to avoid mode coupling between the adjacent waveguides seemed to be a losing battle thus the initial previous 5-way structure was implemented in the proposed device.

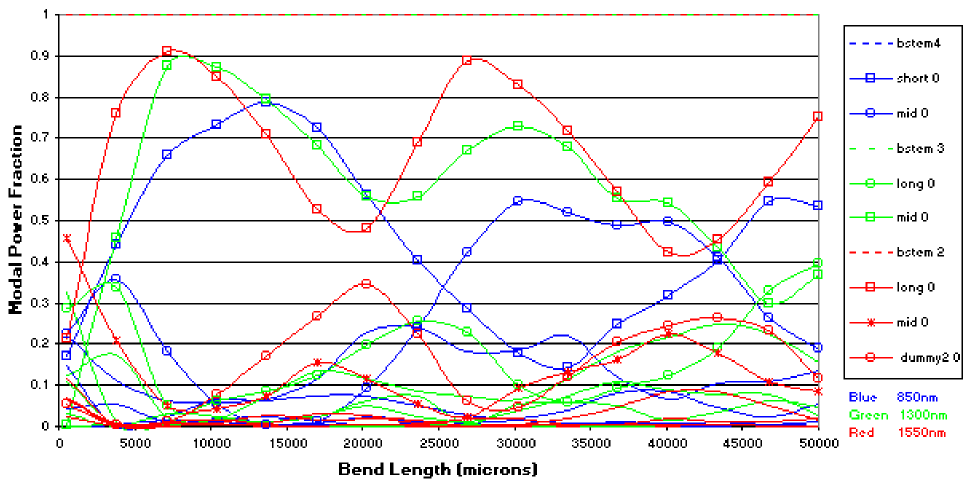



**Figure 4.7:** Adiabaticity as a function of bend length for the dummy arms separating the short, mid and long arms.

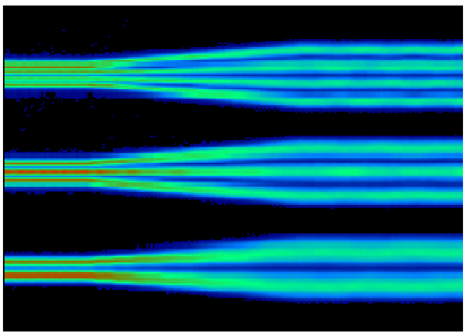
## 4.2 Adiabatic Taper Slope

The purpose of a taper in the adiabatic device is to manipulate the stem mode's effective index. Again, this is using the property that the effective index increases with waveguide width. Thus greater flexibility is introduced into the input-stem and stem-output junctions.

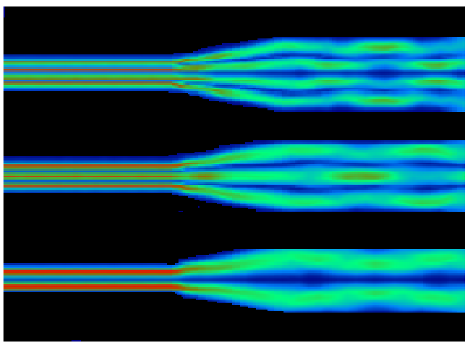
Adiabatic taper gradient was simulated with three duplicate tapers, each having an individual mode input (modes $\Psi_1, \Psi_2$, and $\Psi_3$ from bottom to top) with a constant initial and final width of 5.2 and 20mm respectively, as shown in Figure 4.8. The taper gradient determines the length between the initial and final widths, for example the a bend length of 0.148mm was required for a 1% gradient or the smaller the gradient the longer the taper.

Varying the gradient showed loss to the input mode though mode coupling or radiation was increased in a fluctuating pattern. The cause of fluctuation is explained by the multiple excited modes beating when the taper gradient is large. Thus the recorded modal overlap power fraction is altered, this beating is highlighted in the taper gradient of 0.04 field plot, shown in Figure 4.8.

It is obvious that the taper gradient affects the adiabatic efficiency. Propagating modes are found to be approximately maintained with the 0.01 taper gradient (i.e. 1:100) shown in 4.9(a), allowing a 1% loss to mode coupling or radiation. Also this taper gradient was tested for operation over the telecommunication spectrum in Figure 4.9(b), showing the 1% taper gradient is sufficient in preserving the input mode power to wider (also smaller) taper width. At the 1300 nm wavelength, the $\Psi_3$ mode becomes cut off, and this increases the fraction of input power in the remaining $\Psi_1$ and $\Psi_2$ modes to 50%. Due to the mode being cut off, radiation then reduces the power at the output in the wavelength range 1300-1400nm.

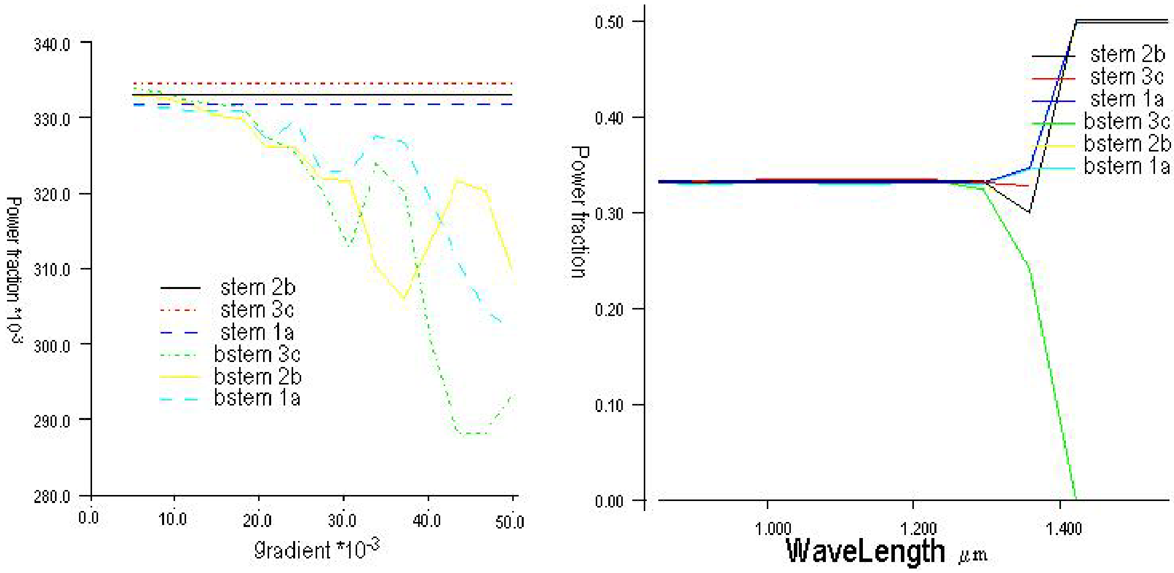


(a) Taper gradient of 0.01



(b) Taper gradient of 0.04

**Figure 4.8**: Optimised taper gradient of 1% and 4% field plots.




(a) Taper gradient

(b) 1% taper spectrum

**Figure 4.9:** Power spectrum of the input and output of a taper as a function taper gradient for modes $\Psi_1, \Psi_2$ and $\Psi_3$. The optimised taper gradient spectrum of 1% shown.

Chapter 5

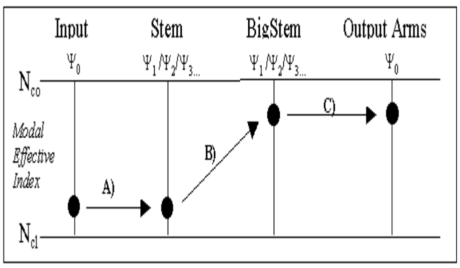
# Devices

## 5.1 How the CWDM Device Works

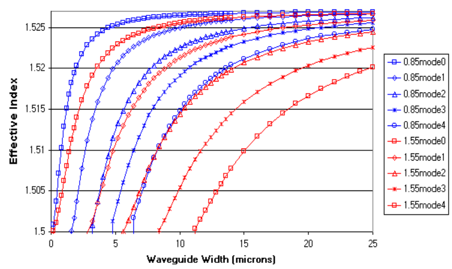
The foundation of the proposed CWDM device is an approximate adiabatic mode evolution. The key concept is that a super-mode adiabatically evolves through an n-way junction. We do not employ coupling or interference phenomena. Power in the super-mode (i.e. the mode of the whole wide structure) is redistributed to appropriate local propagating output modes, with the closest match of effective index to the super-mode [7]-[9]. In terms of power distribution, a single super-mode can take the form of various modes of local waveguide arms as it propagates along a variable waveguide. We take advantage of this to channel the light into the appropriate arm at a given wavelength. In such a case, the power can be concentrated into one mode of one constituent output arm.

A relatively high refractive index difference is required for this type of device to work well e.g. 0.027. Indeed much larger differences (up to 0.4) have been achieved with photosensitive chalcogenide glass waveguides [9] while telecommunication fibre refractive index difference are usually the order of 0.006.

How can the adiabatic processes be used to manipulate modal effective indices to produce a useful device? Here we propose a method, depicted in Figure 5.1(a). Stages A, B and C are discussed below. For convenience a simplified modal effective index characterisation plot is shown in Figure 5.1(b)




(a) Method of modal effective index manipulation.




(b) Modal effective index as a function of waveguide width for the wavelength case of 850nm and 1550nm.

**Figure 5.1**: Device operation

A Demultiplexed wavelengths evolve from the fundamental mode into higher-order modes, depending on the wavelength. This process becomes adiabatic with the input arm gradually met by a supplementary or phantom input arm, with the sum of the input and phantom arms equalling the stem width. The input width is small, so that the effective index is near the cladding index.

B An adiabatic taper expands the waveguide width such that the propagating modes are maintained but their effective indices increase towards the core effective index. This results in the demultiplexed wavelength modes having approximately the same effective index values.

C The demultiplexed wavelengths evolve from the higher-order modes into fundamental modes of the multiple demultiplexer output arms. The selected wavelengths are driven to the output arms with the closest effective index. The output arm widths are largely independent since the modes of the wide part of the stem have approximately equal effective index values. Also, at constant effective index values (i.e. near core index), the fundamental mode at different wavelengths has width differences of the order of half a micron.

## 5.2 Multiplexing and Demultiplexing Devices

Device operation for 2 and 3 wavelength bi-directional de/multiplexing, using the mentioned method, are detailed below. Thus far demultiplexing has been the prime focus for bend lengths consideration, this is due to the difficulty of spectral filtering in the demultiplexer compared to the spectral combiner operation of the multiplexer. This in mind the multiplexing operation requires a overview of input dimensions. The input dimensions of the phantom input arm needs to have stringent conditions placed upon it for bi-directional operation. Reasons for this is that phantom arm should not act as a source of loss for multiplexing wavelengths. Therefore there is to be no approximate effective index match to any propagating operational wavelength stem mode. This condition may require additional phantom inputs.

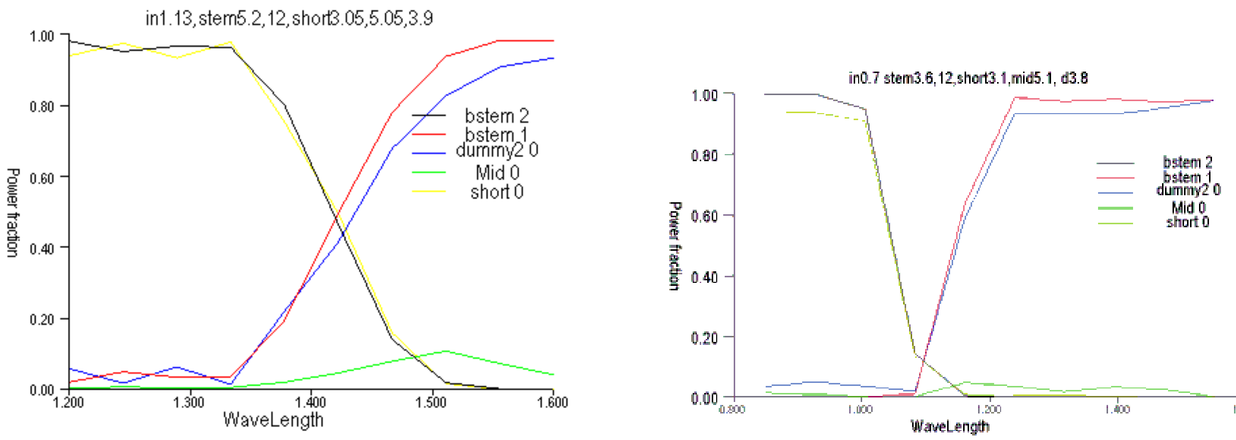
### 5.2.1 2 Wavelength Demultiplexer

The 2 wavelength demultiplexer structure is based on Figure 2.1. At the input arm, the demultiplexing wavelength modes are assumed to be $\Psi_0$. Mode power evolves into modes $\Psi_1$ and $\Psi_2$ in the stem, where the power in the two excited modes depends on wavelength. The stem width is tapered to a larger width, the final stem modal power fraction for the $\Psi_1$ and $\Psi_2$ modes is plotted as Bstem 1 and Bstem 2 respectively in Figure 5.3.

The output of the short width is selected to have a close modal effective index match with the stem $\Psi_2$ mode and output $\Psi_0$ mode. Similarly, the longer wavelength stem $\Psi_1$ and output $\Psi_0$ modal effective index is matched by the 'dummy2' width. The 'Mid' width is used to maintain the adiabatic transition from the 'Bigstem' width to the output arms by satisfying the condition that the sum of the output arm widths equal the 'Bigstem'


**Figure 5.2**: 2 wavelength CWDM structure.

width. The two examples show small parameter changes in the input and stem waveguide widths, and these are followed by optimisation of output arm widths.




(a) Demultiplexer type 1 dimensions are input 1.13, stem 5.2, bigstem 12, short 3.05, mid 5.05 and dummy2 3.9.

(b) Demultiplexer type 2 dimensions input 0.7, stem 3.6, bigstem 12, short 3.1, mid 5.1 and dummy2 3.8.

**Figure 5.3**: Complete 2 wavelength CWDM devices.

With the input cutoff conditions of a small input width and a stem width smaller than a cut-off width for a particular mode, it is noted that the cross-over point occurs at a low power fraction, as the cross-over wavelengths are converted into radiation. This cutoff effect could be exploited to reduce cross-talk. Alternatively, the cutoff approach could be used to create variable ratio (x:y) coupler devices dependant on wavelength.

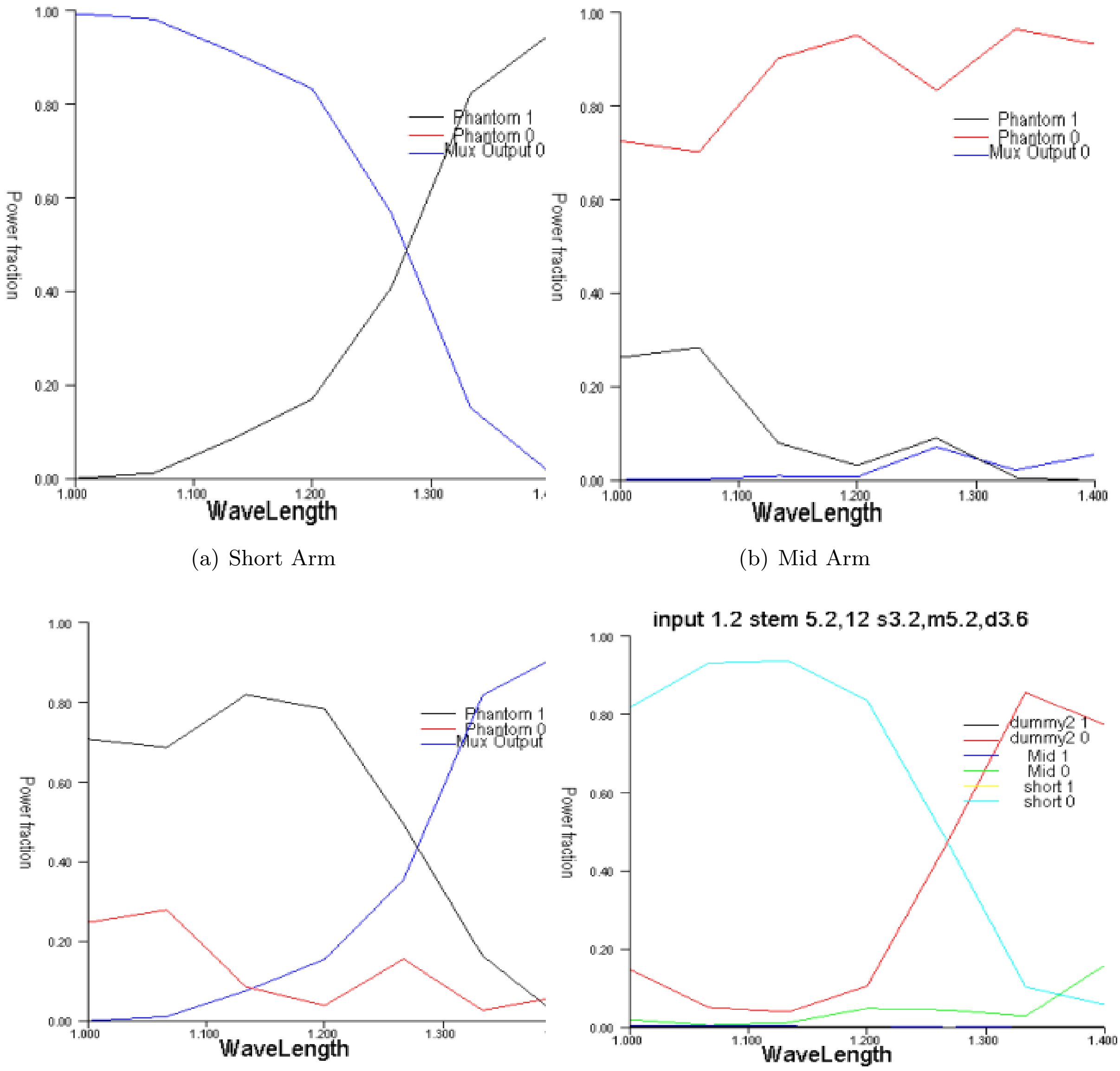
### 5.2.2 2 Wavelength Multiplexer Properties

Here the bi-directional operation of the de/multiplexer device are examined. The 1100nm and 1350nm wavelength are effectively operating in both directions as seen in figure 5.2, where input power to the short, mid and dummy arms are simulated over the 800-1600nm wavelength range and the output power in the 'mux' and phantom output arms are recorded. In the demultiplexing direction the short and dummy2 arms are demultiplex 1100 and 1350nm channels respectively. Interesting to note in the multiplexing direction that any input in the middle arm has the majority of power converted into the phantom arm. Also the phantom arm is coupled into when the wrong wavelength is sent down one of the short or dummy2 arms.

The multiplexing spectrum could only be simulated one arm at a time due to phase conditions of super-modes. In a multiplexing device it is assumed that each input arm carries one unique wavelength range or channel. The simulation monitors the phantom modes and the multiplexing output fundamental mode. In the short multiplexing input arm, almost 100% of modal power is evolved into the multiplexing output arm at 1000nm. Then the modal power fraction declines to zero at 1400nm. The remaining power is evolved into the phantom's $\Psi_1$ mode. When the input is tested at the mid arm both phantom modes are evolved into, but primarily the $\Psi_0$ mode. Less then 10% of the multiplexing output is evolved into at 1270nm which corresponds to the cross over point of the stem mode, thus as designed the mid arm has no real effect on multiplexing. Phantom modes are again evolved when the dummy input is tested, but only in the lower spectrum. Less then 10% of phantom modes are evolved into above 1350nm. The multiplexer output arm is evolved into from 1100nm on, only reaching 80% at 1350nm and a max at 1400nm of about 90%.

Comparing the multiplexing spectrums to the demultiplexing spectrum a few questions are aroused. Like is reciprocity fulfilled for the short and dummy arms and is cross talk an issue in this device? Reciprocity can be assumed if the input power is the same in both directions and the same output power is recorded. It is then observed that the multiplexing spectrum only approximately fills the reciprocity condition. Reason for this may be a result of the input mode having a slightly different shape to that of the reverse direction output.

To clarify this further consider the 1.066$\mu$m wavelength. In the demultiplexing direction the short arm the output power is 96.4% and also the dummy arm is 2.9%. In the multiplexing direction the short input contributes to the 'mux output' 99% while the dummy arm contribution is 0.6%, therefore the mux and demux results are approximately equal.




(a) Short Arm

(b) Mid Arm

(c) Dummy Arm

(d) Demultiplexing spectrum

**Figure 5.4:** 2 wavelength Multiplexing and Demultiplexing device spectrum. The structure dimensions being Short 3.2, Mid 5.2, Dummy2 3.6, Bigstem 12, stem 5.2, phantom 4, and mux output 1.2.

### 5.2.3 3 Wavelength Demultiplexer

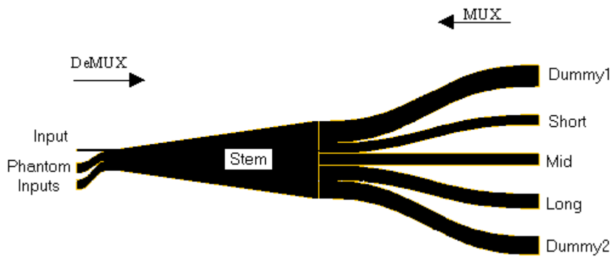



**Figure 5.5**: 3 wavelength 5-way CWDM structure.

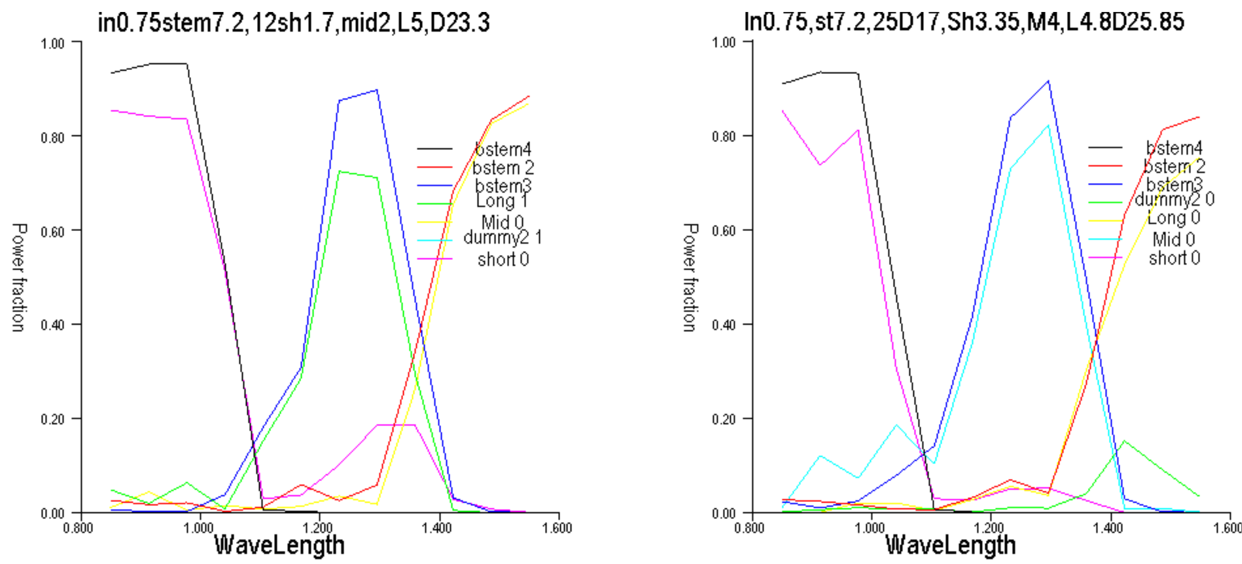
The operation follows the same idea as the 2 wavelength Demultiplexer with some exceptions. Modes $\Psi_2, \Psi_3$ and $\Psi_4$ are excited in the stem, but the stem width is tapered to 12 and 25 $\mu$m for the respective 4- and 5-way junction cases. The structure for the 5-way device is based on Figure 5.5, which is the 4-way device apart from one less output arm. The evolved output modes in the 4-way junction case are $\Psi_0$ for 850 nm and 1550nm wavelengths, while $\Psi_1$ mode evolved into at 1300nm wavelength. The 5-way junction modal effective index in the Bigstem are close together, allowing each output to evolve into a $\Psi_0$ mode. The efficiency at 1550nm is approximately 80% and the $\Psi_2$ stem mode evolution efficiency of 85%. These are limited by the efficiency of the adiabatic evolution through both the input to stem and bigstem to output junctions.

The spectra in Figure 5.6 show the modal power fraction at the end of stem (Bstem)and at the output arms (Dummy1, Short, Mid, Long and Dummy2). Both spectra are similar due to the same input-stem junction conditions, but it can be noted that crosstalk is about 4 time higher in the 4-way device at 1300nm. Also in the 5-way device, one of the dummy arms act as a leakage between 1400-1550nm.

### 5.2.4 3 Wavelength Multiplexing Properties

Operating the device as a multiplexer brings into play the 2-way input-stem junction, where the phantom arm can be a source of power leakage. When the stem width is much greater then the input width, modes do not evolve into the 'mux output' arm, due to the phantom arm approximating the stem width or having effective index matches to higher order modes. While this does not affect the demultiplexing operation in the multiplexing mode, reciprocity fails with a 2-way junction. Considered here is the 4-way device.

The phantom arm having almost the same width as the stem, has the expected result of power continuing into the phantom arm, due to close match of higher order effective




(a) Spectrum 4-way Demultiplexer. The structure dimensions are input 0.75 , stem 7.2, bigstem 12, short 1.7, mid 2, long 5 and 'dummy 2' 3.3.

(b) 5-way Demultiplexer Spectrum. The structure dimensions are input 0.75, stem 7.2, bigstem 25, 'dummy1' 7, short 3.35, mid 4, long 4.8 and 'dummy2 ' 5.85.

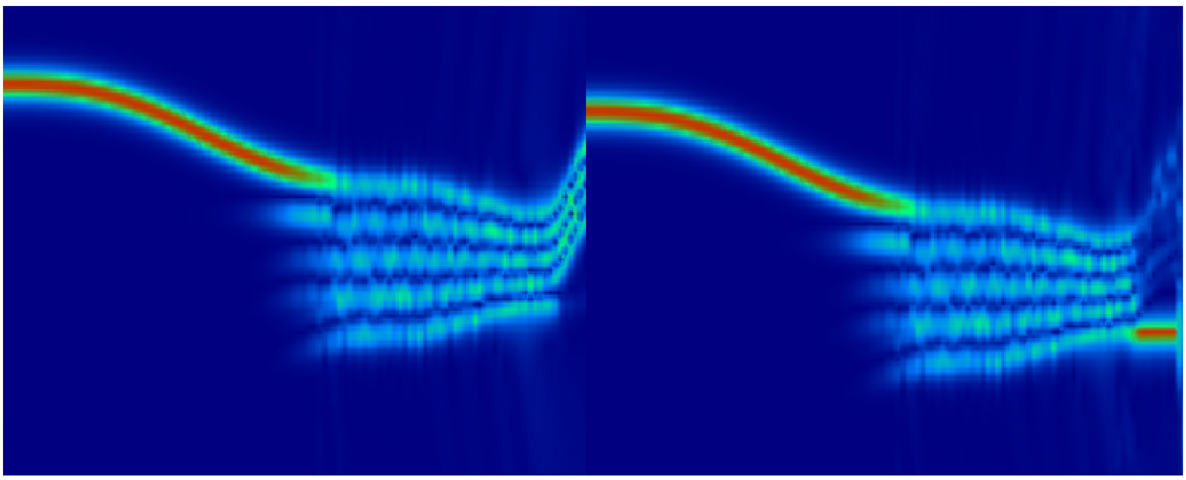
**Figure 5.6**: Complete 3 wavelength CWDM devices.

index. Solution to avoid this was to add an additional phantom arm. An example is shown in Figure 5.7, showing both output results for the same multiplexing device operating at 850nm but having either one or two phantom arms. The two phantom arm gives more flexibility in driving the wavelength into the mux output arm, while the single phantom arm case does not evolve any power into the mux output arm.

The input-stem junction was converted from the Y-junction to a 3-way junction. Where the two phantom arms now had the same widths, reciprocity was restored and the mux output arm returned to the closest modal match for the multiplexing wavelengths. The spectrum sweep at each mux inputs produced Figures 5.8 and 5.9.

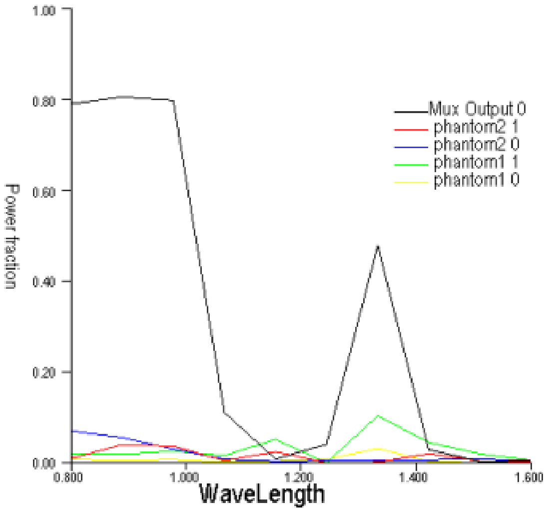
Multiplexing of 3 wavelengths at 800-1000nm, 1220-1300nm and 1450-1600nm ranges for the respective short, mid and long input arms with approximately 80% efficiency was simulated. Similarly to the 2 wavelength multiplexing case the wrong input wavelengths generally evolve into the phantom arms. Although there exists a small amount of potential crosstalk. The short input in the 1300-1400nm region and the long arm begins to increase modal power into the 'mux output' from 1300nm onwards, both causing crosstalk to the mid arm. Ideally crosstalk in the multiplexing direction would not be a problem as each input would only have the appropriate operating channel.

There exist other possible input conditions such as dummy input modes and the fundamental input of the long arm, these are characterised in Figure 5.9. The single mode propagation in the Long arm continues to propagate into the phantom arms across spectrum. This is similar to the Dummy single mode propagation. The Dummy $\Psi_1$ mode propagation turns into a highly radiative modes (i.e. mode is not bound).

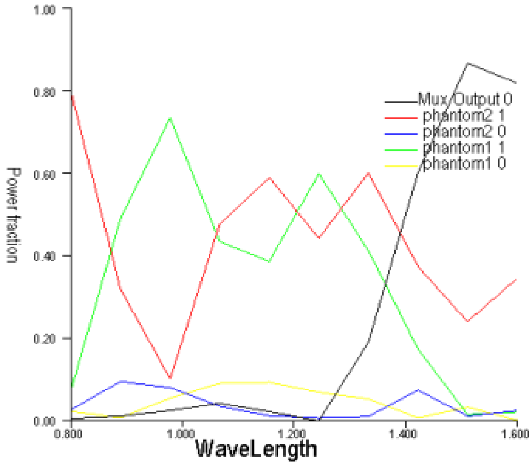


(a) One Phantom Arm (b) Two Phantom Arms

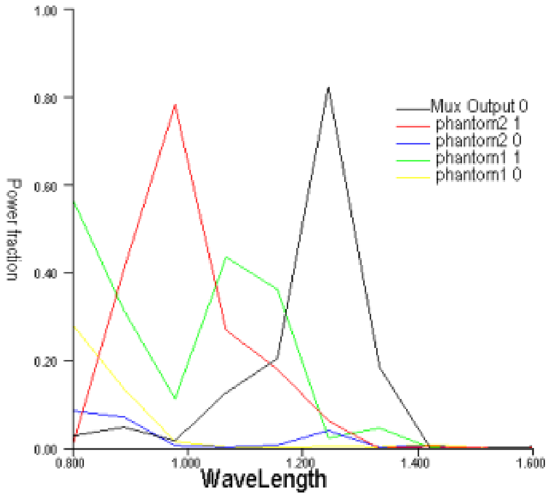
**Figure 5.7:** Multiplexing field operation of the 4-way device at 850nm with a) only one phantom arm, b) two phantom arms.




(a) Short $\Psi_0$

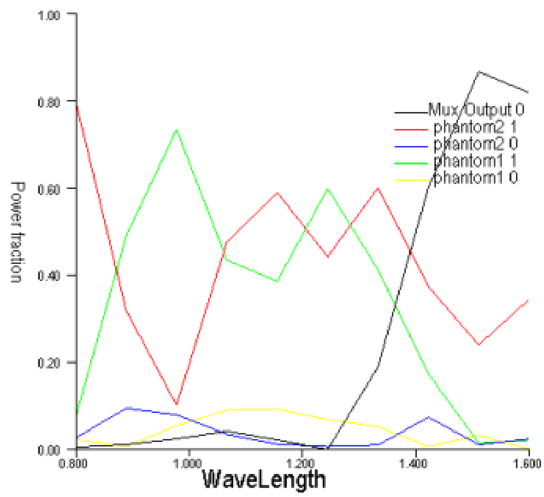



(b) Mid $\Psi_0$

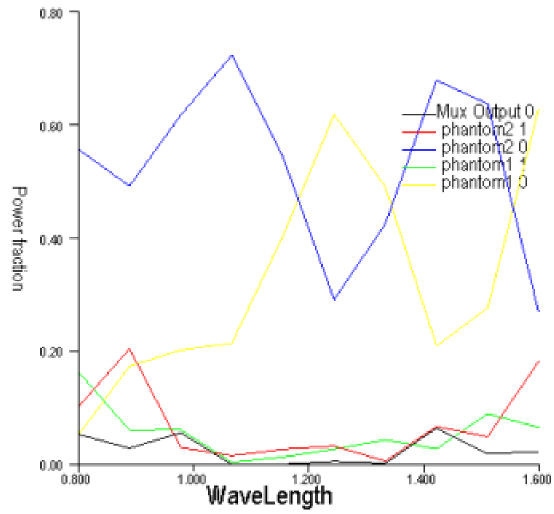



(c) Long $\Psi_1$

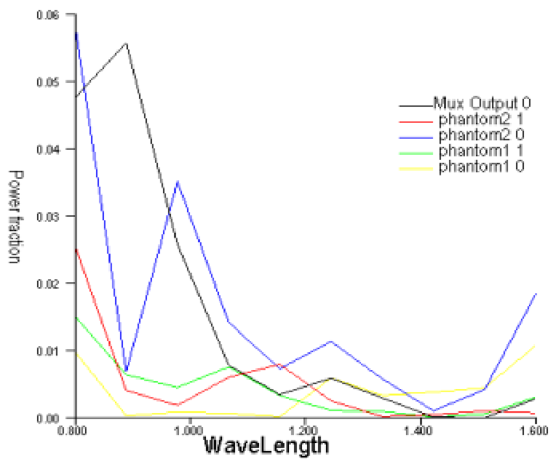
**Figure 5.8**: Multiplexing spectrum for the operational short, mid and long arms




(a) Long $\Psi_0$




(b) Dummy $\Psi_0$

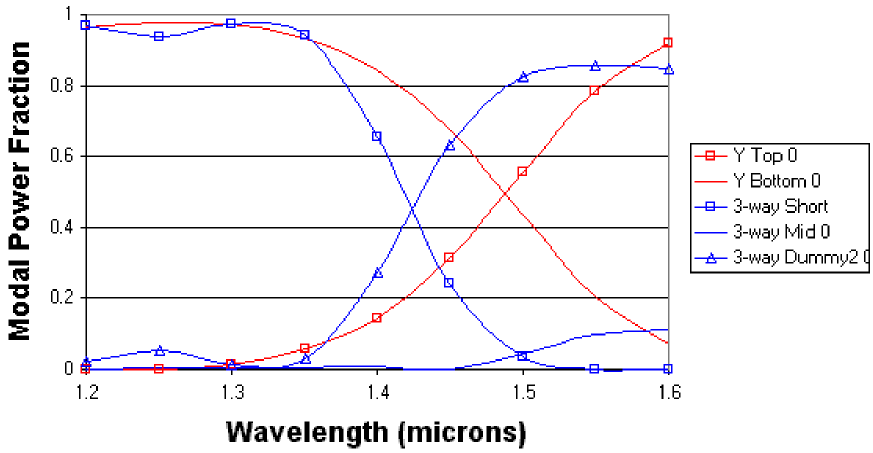



(c) Dummy $\Psi_1$

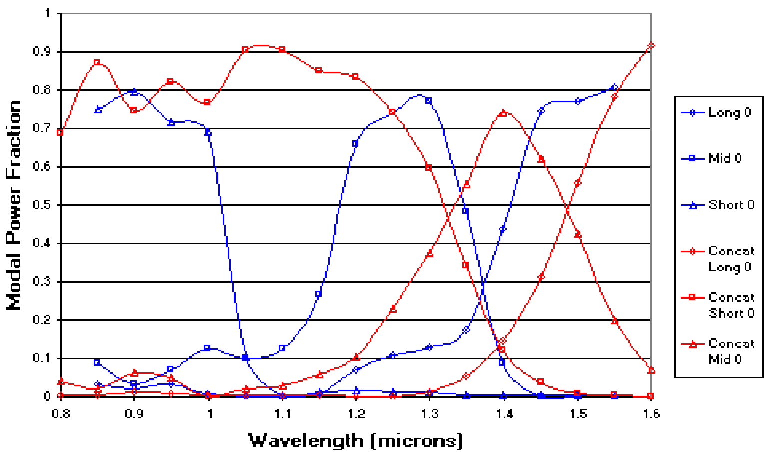
**Figure 5.9**: Multiplexing in redundant modes and dummy arm.

## 5.3 Comparison to Previous Adiabatic Devices

The spectrums of modal power fractions for the two and three wavelength devices are superimposed by Y- and concatenated Y- junctions spectrums respectively. The devices designed in this thesis are represented by blue lines in Fig 5.10. Issues to be compared is the wavelength filtering, crosstalk, insertion loss and device length. The refractive index difference values are not equal for the both devices, with the Y-junction device difference of 0.04 and the thesis devices have 0.027 difference. This is reflected in the size of the two devices, as higher index difference generate smaller waveguide width structures.




(a) 2 wavelength device




(b) 3 wavelength device

**Figure 5.10:** comparison between a) two wavelength division demultiplexing by a Y- and 3-way junction b) three wavelength division demultiplexing via concatenation of Y- junctions or a 5-way junction.

### 5.3.1 2 Wavelength Device

The Y-junction device is compared to the 3-way type 1, where both 1300 and 1550nm wavelengths are separated.

The 3-way device has narrower switching bandwidth, and thus can switch/demultiplex for smaller wavelength separations. The power throughput is above 95% for the short wavelength in both cases, while at the longer wavelength the 3-way device has a throughput of about 85%, this is caused by modal power leakage in the mid arm. The Y-junction throughput becomes about 90% at 1.6$\mu$m but at 1550nm only 80% throughput is achieved. Cross talk tends to be directly influenced by the throughput of the Y-junction device, as the remaining power is transmitted in the adjacent output arm. Unlike the Y-junction the 3-way junction crosstalk is minimised by the remaining power being channelled into the mid arm. The extra arm and tapered stem in the 3-way junction also acts to bring greater tuning range, this can be seen by the multiple 2 wavelength devices shown in this thesis, some other applications are highlighted in the opimisation section. The size of the Y-junction device is about 3-4mm and the 3-way device is about 10mm. Again the size of the two devices can not really be compared due to the difference in refractive index profiles, but the obvious advantage of the 3-way junction is that only the input and output junctions are required to transform fundamental mode inputs into spectrally selecting output arms that propagate channels in the fundamental mode. The Y-junction demultiplexer has the input and output junctions but an extra junction is required to convert the $\Psi_1$ mode into the fundamental mode.

### 5.3.2 3 Wavelength Device

The concatenated Y junction device is compared to the 5-way device. Although the two devices differ in the wavelengths that are demultiplexed substantial comparisons can be made.

5-way junctions demultiplexes of 1550, 1300 and 900nm at approximately 80% with about 150 to 200nm spectral filter bandwidths. The concatenate Y junction demultiplexes 1550, 1400 and 850 to 1200nm wavelengths, the spectral filter bandwidths vary greatly. The demultiplexing process for the Y-junction case is limited by the first Y-junction, i.e. the highest wavelength is separated it the fundamental mode of the top output while the remaining wavelengths need to evolve into the $\Psi_1$ mode in the second stem section. This limits the potential filtering of the middle wavelength with minimal crosstalk. While the middle wavelength in the 5-way junction has cross talk, it is at least half that of the Y-junction device and also only leaks into one channel output rather than 2. Again the flexibility of the tapered stem plus dummy arms allows the 5-way junction to be easily tuned, provided wavelength separation is greater then 150nm between pass band edges. The length of the 5 way junction is about double the concatenated Y-junction, with the size of the Y-junction device being about 4-5mm and the 5-way device is about 10mm, but remember the Y-junction has a higher refractive index difference advantage.

## 5.4 Optical Add/Drop Multiplexer Proposal

Add/drop multiplexers pass all the input wavelengths except for one spectral region/channel that is tapped out. This tapped out channel is referred to as being 'dropped'. The 'added' channel has the same spectral region of the dropped channel but it carries totally independent information or modulated bits. Thus it is important that the dropped channel does not pass through the device, as this would become homodyne crosstalk with the added channel.

How then can an add/drop multiplexer be made with the geometrical device already proposed in this thesis. The 5-way device could be used to initially demultiplex the input spectrum where one arm is directs one wavelength to the dropped receiver or network. A slightly altered 5-way multiplexer could then be connected. The alteration would be to reorder the multiplexer input arms such that the added and dropped channel arma are free to extend to the extremities of the device, and the other arms are directly connected to the correct demultiplexed channel. This method could be possible but it is has the disadvantage of high design complexity. An alternative add/drop multiplexer may use a 3 or 4 way junction, and also the phantom arm could also be used at the added channel input, with the condition that the excited stem mode is different to the dropped stem mode.

Chapter 6

# Device Tolerance

Tolerance needs to be considered for realistic operation and manufacturing capabilities. Also tolerance shows what effect a change in waveguide width or refractive index has on the performance characteristic of the device.

Design tolerance is important for knowing how effectively the device can be produced with the limitations in resolution and accuracy of waveguide development processes. It also relates to the versatility of the device in its ability to de/multiplex broad wavelength ranges, operate by sources with wavelength drifts, or new applications of switching/sensing via refractive index changes may be realised.

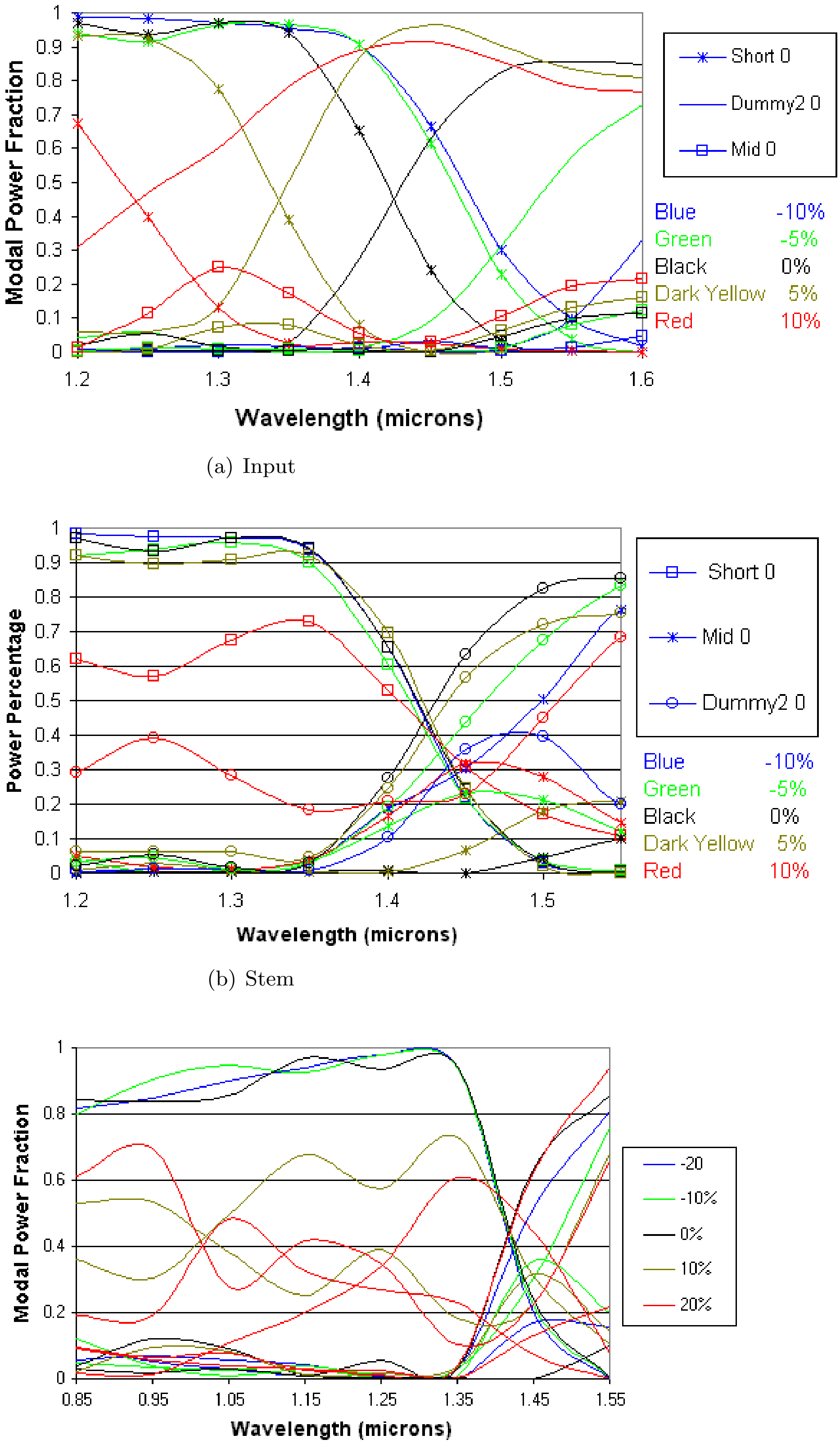
To test the tolerances, design parameters are selected and a percentage change to the value is controlled by a new control parameter tolerance(%). Then simulation is run with a sweep of tolerance values. The input, stem, bigstem and short waveguide widths, and also refractive index difference, are the device design parameters being studied. The device is simulated over the operation spectrum with the changed parameter being altered as a percentage of the initial value designed for the device.

## 6.1 2 Wavelength Device Tolerances

Increasing the input width causes the lower order modes to be favoured, shown in modal manipulation section. Shifts for the 1300nm and 1550nm device to operate requires a tolerance ±3% in the waveguide width, determine in Figure 6.1(a). A similar result was determined for the 850 and 1550nm device with the input tolerance being ± 5-6%. The slight increase in tolerance is afforded due to the large wavelength separation.

The Bigstem-Output junction tolerance is dominated by the output fundamental mode effective index sensitivity to waveguide width. The challenge is to arrange the output widths to have effective indices that match the fundamental modes of the demultiplexed wavelengths and also have no close match to the other modes. In Figure 6.1(b), for 5% in bigstem width, the device remains operational, while at 10%, the change in modal effective index is sufficient for the other output arms to have competing effective index matches. At -10%, the device would still operate, but the 1550 nm light would exit through the Mid output arm, and for +10%, the Mid arm starts competing for the 850nm light.

The 2 wavelength demultiplexer was also investigated for refractive index difference changes, shown in Figure 6.1(c). It was found that the 1300 and 1550nm device was sensitive to increases in refractive index difference changes, primarily due to the constant sized waveguide effective index becoming approximately equal to the core. Leading to both outer arms of similar width having approximately equal effective index values. On the other hand decreasing the refractive index allows for the two operational outputs to be distinguishable. The device could still operate with a decrease of refractive index deference of up to 20%, although leakage into the mid arm becomes an issue at +20% tolerance values.




(a) Input



(b) Stem

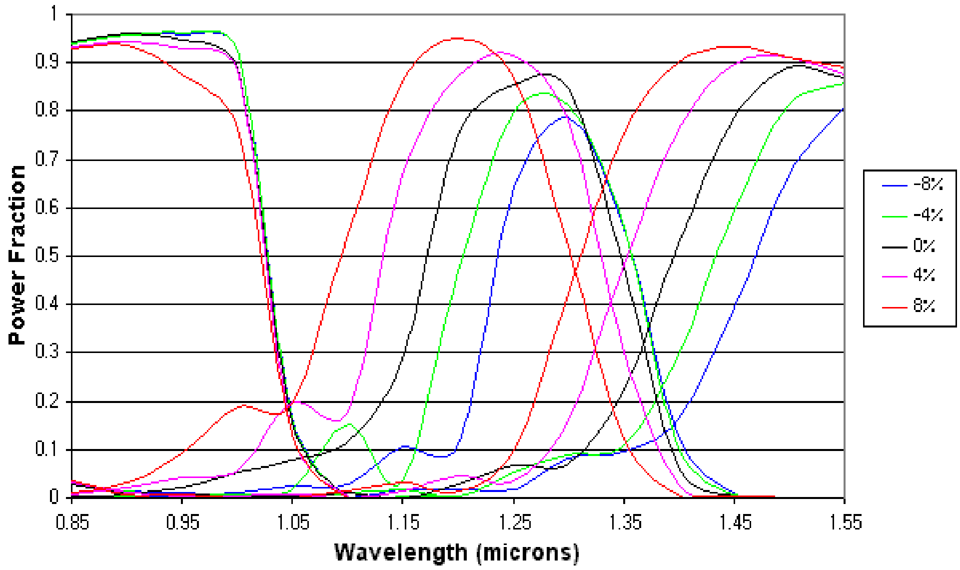


(c) Refractive Index

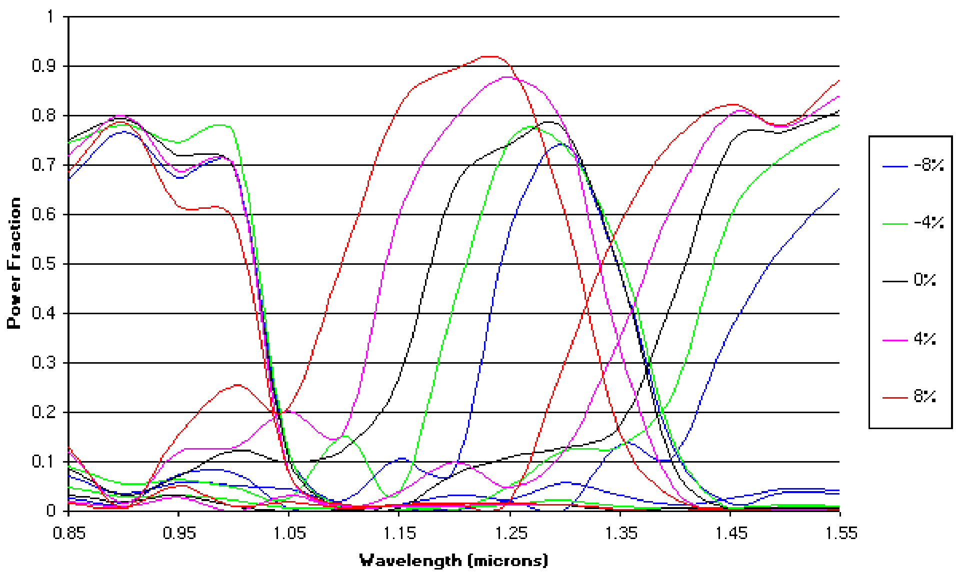
**Figure 6.1**: Tolerance in the two wavelength demultiplexing type 1 device.

## 6.2 5 Arm Device Tolerances

It was found that the input waveguide width significantly affects the evolution of the input wavelengths into the stem mode, shown in Figure 6.2. With a 5% change in width, the cross-over wavelength for the multiple stem modes is changed. The shift is reduced for higher-order modes. An increase in input width shifts the stem modes to lower wavelength and also increases the filtering bandwidth. The output spectrum also follows the stem mode trend with no significant changes to the filter shapes. The stem width can affect the input stem modes, but for a 5% change in width , there is no significant effect on operation.




(a) Tolerance to input width seen at the stem.

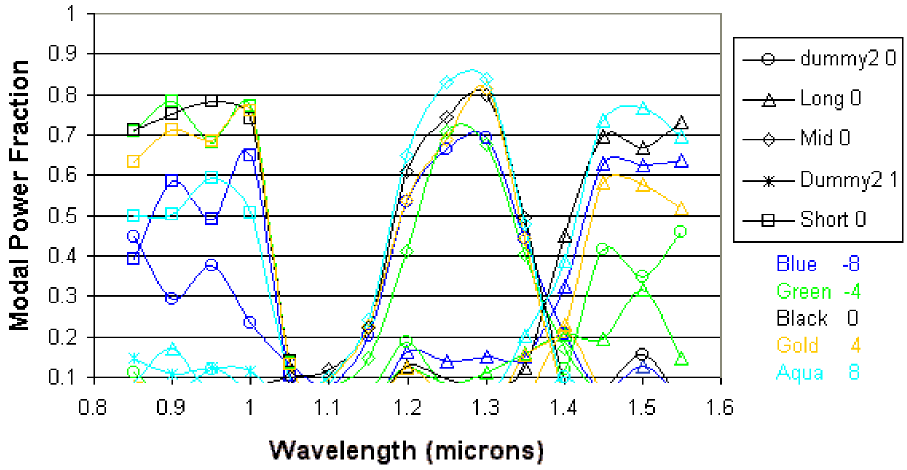



(b) tolerance to input width seen by the outputs.

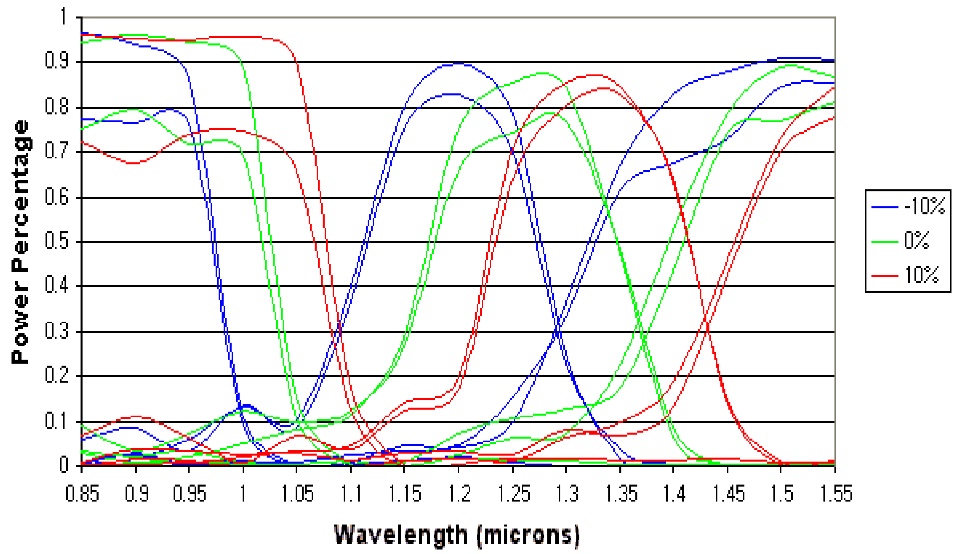
**Figure 6.2**: Input tolerance for the 5-way device

The bigstem width in the 5-way device is investigated in Figure 6.3(a). The trend observed was that in increasing the bigstem width, the device still remained operational, although an optimum uniform peak power for all three channels is not achieved. In decreasing the bigstem width the dummy2 arm begin to dominate. This reflects the dummy2 arm becoming smaller due to the condition that the $\sum$ of outputs equals the stem output width. This effect could be alleviated by evenly reducing both dummy arms, but eventually the dummy arms would become dominant. Here the Bigstem width tolerance is an increase of 10%. To make the Bigstem width increase it's tolerance one only needs to design the device for wider bigstem, but this would require longer tapers.

Tolerance in refractive index difference is also investigated, noting that 10% accuracy is well within existing fabrication techniques. In Figure 6.3(b)an increase in the refractive index difference is shown to shift the spectrum to longer wavelengths. Considering the 3 wavelength 5-way demultiplexer, a negative 10% change in the refractive index difference produces a shift to lower wavelengths by approximately 100nm, while a 10% increase gives approximately a 50nm shift to longer wavelengths. This could indicate that device tolerance could be smaller at higher refractive index values, although the waveguide widths would also be decreased.




(a) Tolerance to outputs via change in bigstem.




(b) Tolerance for the 5-way 3wavelength CWDM device tolerance to refractive index difference, with a of ± 10% change.

**Figure 6.3:** Tolerance of 5-way device with respect to Bigstem width and refractive index difference

Chapter 7

# Conclusions

With the objectives in mind, the initial idea was to have the Y-junction connected to a 3-way junction, that evolved the input fundamental modes into the $\Psi_0, \Psi_1$ or $\Psi_2$ output modes, dependant on wavelength. Then Y-junctions would convert high-order modes to the fundamental mode. Modal effective index was characterised as a function of waveguide width for the telecommunication wavelengths 850, 1300, and 1550nm by using the Scalar Wave Equation, eigenvalue equation for weak-guiding planar slab waveguides. Maple was used to calculate results in a spread sheet format. Once the modal effective index was plotted it was quickly realised that the initial idea would not work, due to adiabatic conditions failing as the sum of output arms widths would need to be larger then the stem width.

Back at the drawing board, can any device work with a 3-way junction? A 2-wavelength demultiplexer was formulated that evolves both channel outputs directly into the fundamental mode. The additional arm was used to maintain an adiabatic transition. Problem to be considered with an additional output arm is that power leakage to the dummy arm needs to be minimal for the operational channels, thus complexity in design is increased.

The creation of this new 2 wavelength demultiplexer changed the direction of the project, for a more effective 3-wavelength demultiplexing device was now envisaged. The outputs would all be in the fundamental mode! But how many output arms would be required to do this 4, 5 or 6? Now as the number, $n$, of output arms increased the length of the adiabatic device was expected to increase as $n^4$. This problem of adiabatic bend lengths was investigated and it was found that increasing the number of output arms does not significantly increase the bend length, this is provided that the devices with larger output arms have higher propagating mode orders. As adiabatice bend lengths is smaller for higher order modes.

Another realisation was that the stem width does not need to be fixed. This design relaxation has the advantage utilising particular modal effective index properties, for example the large waveguide widths locate wavelengths with different modes to a similar $n_e$ value. With this in mind the input-stem junction can be optimised without compromising the stem-output junction operation. While the input and output of the stem are not equal the issue of taper gradient needed to be addressed. The taper gradient has to be as short as possible while still maintaining adiabaticity, 1% taper gradient was found to be sufficient.

The input-stem junction was optimised for all wavelengths in one mode and combinations of all different modes. Tested different output stem width and coming to the conclusion that the output stem width was found to be most useful at larger values, if the outputs need to be in the fundamental mode.

A three wavelength demultiplexing device which has two outputs propagating in the fundamental mode and the third output propagating in the $\Psi_1$ mode was developed. With the same input-stem condition but increasing the output stem width an improved device was developed, that allowed all outputs channels to propagate into the fundamental output mode, but this required an additional dummy arm. The final device operates in CWDM regime with a minimal number of junctions and only adiabatic processes employed. Although the multiplexing ability of the final device has yet been developed, although the 3- and 4-way device show that multiplexing operational can be acheived. Thus making the devices bi-directional in the sense that, in one propagating direction the device will demultiplex and the reverse direction multiplexing of light will occur.

The tolerance was then studied for the demultiplexing characteristic of the developed 2- and 3- way junction devices by varying design parameters. Highlighting the importance manufacturing accuracy with waveguide width deviations of 5% in the smallest width (input) can make the device fail to work at the designed wavelengths. Thermal tuning capabilities could be possible in the 5-way device where temperature can change the refractive index properties. This reached the primary goals of the thesis.

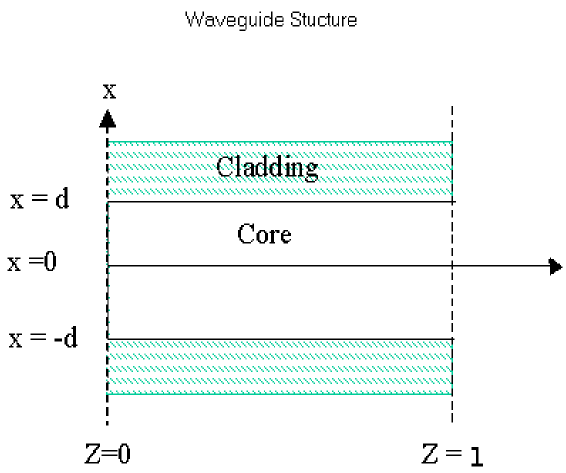
The feasabliity of making the devices in the proposed RPO manufacturing process is low, due to the resolution waveguides can be accurately made for widths of about 4$\mu$m up. As the smallest waveguide with are 0.75-1.2$\mu$m other higher resolution waveguide fabrication techniques need to be used.

Chapter 8

# Appendix

## 8.1 Slab Eigenvalue Equation

With the boundary conditions considered for the scalar wave equation in 3.1.2, the eigenvalues can be derived as follows.




$\Psi cl$ *is the Cladding Wave function* ; $\Psi co$ *is the Core Wave function*

$$\Psi cl\left(\frac{|x|}{r}\right) = \frac{e^{-W\frac{|x|}{r}}}{e^{-W}} \qquad at\ \Psi cl(1) = 1$$

$$\Psi' cl\left(\frac{|x|}{r}\right) = \frac{-Win}{r}\frac{e^{-W\frac{|x|}{r}}}{e^{-W}} \qquad at\ \Psi' cl(1) = -W$$

$$\Psi co\left(\frac{|x|}{r}\right) = \frac{Cos(U\frac{|x|}{r})}{Cos(U)} \qquad at\ \Psi co(1) = 1$$

$$\Psi' co\left(\frac{|x|}{r}\right) = \frac{-U\ Sin(U\frac{|x|}{r})}{Cos(U)} \qquad at\ \Psi' co(1) = -U\ \frac{Sin(U)}{Cos(U)}$$

$$= -U\ Tan(U)$$

$\therefore$ *at boundary for even Modes*

$$W = U\ Tan(U) \qquad eigenvalue\ equation$$

$$\Psi cl\left(\frac{|x|}{r}\right) = \frac{e^{-W\frac{|x|}{r}}}{e^{-W}} \qquad at\ \Psi cl(1) = 1$$

$$\Psi' cl\left(\frac{|x|}{r}\right) = \frac{-W}{r}\,\frac{e^{-W\frac{|x|}{r}}}{e^{-W}} \qquad at\ \Psi' cl(1) = -W$$

$$\Psi co\left(\frac{|x|}{r}\right) = \frac{Sin(U\ \frac{|x|}{r})}{Sin(U\ )} \qquad at\ \Psi co(1) = 1$$

$$\Psi' co\left(\frac{|x|}{r}\right) = \frac{U\ Cos(U\ \frac{|x|}{r})}{Sin(U\ )} \qquad at\ \Psi' co(1) = U\ \frac{Cos(U\ )}{Sin(U\ )}$$

$$= U\ \ Cot(U\ )$$

$\therefore$ *at boundary for Odd Modes*

$$W\ \ = -U\ Cot(U\ ) \qquad eigenvalue\ equation$$

## 8.2 Maple Scalar Wave Equation Code

```
Beta Calculator Written by Andrew Molloy April 2004 Start
> restart;
Varables
>
Wavelenght (lambda)    ***com 1550nm, 1300nm lambda1:= 1.5;

half widths of arms microns
> pstem:=5;
   nco1:= 1.527;                                   Refractive index of core
  ncl1:=1.5;
deln1:= nco1 -ncl1;
                       deln is the change in refractive index from the core to cladding

Calculations Wavenumber,
> k:=lambda-> evalf(2*Pi/lambda);
v:=(p,lambda,nco,deln)-> k(lambda)*p*sqrt(2*nco*deln);

Propagation constant nth Modes
> beta:=(n,p,lambda,nco,deln)-> evalf(sqrt(k(lambda)^2*nco^2-u[n](p,lambda,nco,deln)^2/(p^2)));

Solution to u(n)

Example using the above parameters: boundary condition solution
for the wavefunction at cladding core interface of planar
waveguide First plot has u1 units in terms of pi and the second
one is the actual value.

  plot([u1_pi*Pi*tan(u1_pi*Pi),sqrt(v(pstem,lambda1,nco1,deln1)^2-(u1_pi*Pi)^2)]
  ,u1_pi=-1.5*v(pstem,lambda1,nco1,deln1)/Pi..1.5*v(pstem,lambda1,nco1,deln1)/Pi
  ,-15..15);plot([-u1_pi*Pi*cot(u1_pi*Pi),sqrt(v(pstem,lambda1,nco1,deln1)^2-
  (u1_pi*Pi)^2)],u1_pi=-1.5*v(pstem,lambda1,nco1,deln1)/Pi..1.5*v(pstem,lambda1,
  nco1,deln1)/Pi,-15..15);

>
u[0]:=(p,lambda,nco,deln)->fsolve(u*tan(u)=sqrt(v(p,lambda,nco,deln)^2-u^2),u,0..Pi/2);
u[1]:=(p,lambda,nco,deln)->fsolve(-u*cot(u)=sqrt(v(p,lambda,nco,deln)^2-u^2),u,Pi/2..Pi);
u[2]:=(p,lambda,nco,deln)->fsolve(u*tan(u)=sqrt(v(p,lambda,nco,deln)^2-u^2),u,Pi..3/2*Pi);
u[3]:=(p,lambda,nco,deln)->fsolve(-u*cot(u)=sqrt(v(p,lambda,nco,deln)^2-u^2),u,3/2*Pi..2*Pi);
u[4]:=(p,lambda,nco,deln)->fsolve(u*tan(u)=sqrt(v(p,lambda,nco,deln)^2-u^2),u,2*Pi..5/2*Pi);


betastem[0]=beta(0,pstem,lambda1,nco1,deln1);betastem[1]=beta(1,pstem,lambda1,nco1,deln1);
betastem[2]=beta(2,pstem,lambda1,nco1,deln1);
betastem[3]=beta(3,pstem,lambda1,nco1,deln1);
betastem[4]=beta(4,pstem,lambda1,nco1,deln1);


>      Psi0:=(p,lambda,nco,deln)-> eval(beta(0,p,lambda,nco,deln)/k(lambda));
Psi1:=(p,lambda,nco,deln)-> eval(beta(1,p,lambda,nco,deln)/k(lambda));
Psi2:=(p,lambda,nco,deln)-> eval(beta(2,p,lambda,nco,deln)/k(lambda));
Psi3:=(p,lambda,nco,deln)-> eval(beta(3,p,lambda,nco,deln)/k(lambda));
Psi4:=(p,lambda,nco,deln)-> eval(beta(4,p,lambda,nco,deln)/k(lambda));
```

## 8.3 Simulation code

### 8.3.1 Supermode

### Forward Y-junction

```
/* BBV written DEV file */ /* Contact us at bbv@bbv.nl */

dev_version 410;
MaskDescript(masksize=5,maskReferencePoint=[0,0]);

Material { MAT "Background" : STATE 1 : BACKGROUND : INDEX
"simDim()==3?1946:nback " : COLOR 255,255,255  : THERM "1","1" :
THOPT "1e-005","100000" : ELECT "1","1" : ELOPT "1e-006","1e-006"
: PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT "Waveguide" :
STATE 1 : INDEX "simDim()==3?19468:ncore " : COLOR 128,255,0  :
THERM "1","1" : THOPT "1e-005","100000" : ELECT "1","1" : ELOPT
"1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT
"start" : STATE 1 : INDEX "simDim()==3?1945568:nstart" : COLOR
255,88,0  : THERM "1","1" : THOPT "1e-005","100000" : ELECT
"1","1" : ELOPT "1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" :
GDSLAYER 0;
 };

layer("System",systemtype,view=off,fill=off,width=65,refPlaneY=0,overlayfactor=1,Emk[15240,0,0],ColorRGB(255,255,255))
{
 };
layer("Waveguide",view=on,fill=off,width=25,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(255,200,0))
{
  XY_CrossSection(Material="Waveguide",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="Waveguide",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=5));
 };
layer("start",view=on,fill=off,width=73,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(205,255,0))
```

```
{
  XY_CrossSection(Material="start",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="start",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=85));
 };

DefaultValue(Lambda=1.55,Width=65,ModeNum=0,Polarisation=TE,BpmOrder=2,viewmaskwindow=off,viewbpmwindow=on,viewmode="Normal",DiscretiseBpm=1e-020);
puts(What=""); listcomment(Text="-----------------Definition of
Device------------------"); deflayer(layer "Waveguide");
variable(Name="nback",Value=1.5,Comment="");
variable(Name="ncore",Value=nback+0.027,Comment="");
variable(Name="nstart",Value=ncore,Comment="");
listcomment(Text="---------------Stem----------------");
variable(Name="bigwidtha",Value=12,Comment="");
variable(Name="bigwidthb",Value=12,Comment="");
variable(Name="bigwidthc",Value=12,Comment="");

listcomment(Text="------------Outputs-------------");
variable(Name="shorta",Value=bigwidtha/2,Comment="");
variable(Name="dummy2a",Value=(bigwidtha/2),Comment="");
variable(Name="shortb",Value=bigwidthb/2,Comment="");
variable(Name="dummy2b",Value=(bigwidthb/2),Comment="");

variable(Name="shortc",Value=bigwidthc/2,Comment="");
variable(Name="dummy2c",Value=bigwidthc/2,Comment="");

variable(Name="armoffset",Value=6,Comment="");
variable(Name="inoffset",Value=50,Comment="");
variable(Name="smode",Value=0,Comment="");
variable(Name="topphase",Value=180,Comment="");
listcomment(Text="---Output length-------");
variable(Name="Ls",Value=5000,Comment="");

puts(What="---------Beta Values---------");
variable(Name="beta0big",Value="modepropml('re','neff',0,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta1big",Value="modepropml('re','neff',1,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta2big",Value="modepropml('re','neff',2,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta3big",Value="modepropml('re','neff',3,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta4big",Value="modepropml('re','neff',4,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta5big",Value="modepropml('re','neff',5,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta0shorta",Value="modepropml('re','neff',0,5,nback,shorta,ncore,5,nback)",Comment="");
variable(Name="beta1shorta",Value="modepropml('re','neff',1,5,nback,shorta,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2a",Value="modepropml('re','neff',0,5,nback,dummy2a,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2a",Value="modepropml('re','neff',1,5,nback,dummy2a,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2a",Value="modepropml('re','neff',2,5,nback,dummy2a,ncore,5,nback)",Comment="");

variable(Name="beta0shortb",Value="modepropml('re','neff',0,5,nback,shortb,ncore,5,nback)",Comment="");
variable(Name="beta1shortb",Value="modepropml('re','neff',1,5,nback,shortb,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2b",Value="modepropml('re','neff',0,5,nback,dummy2b,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2b",Value="modepropml('re','neff',1,5,nback,dummy2b,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2b",Value="modepropml('re','neff',2,5,nback,dummy2b,ncore,5,nback)",Comment="");

variable(Name="beta0shortc",Value="modepropml('re','neff',0,5,nback,shortc,ncore,5,nback)",Comment="");
variable(Name="beta1shortc",Value="modepropml('re','neff',1,5,nback,shortc,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2c",Value="modepropml('re','neff',0,5,nback,dummy2c,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2c",Value="modepropml('re','neff',1,5,nback,dummy2c,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2c",Value="modepropml('re','neff',2,5,nback,dummy2c,ncore,5,nback)",Comment="");

puts(What="Bigstem");
variable(Name="beta1big",Value=iprint(beta1big),Comment="");
variable(Name="beta2big",Value=iprint(beta2big),Comment="");
variable(Name="beta3big",Value=iprint(beta3big),Comment="");
puts(What="Short");
variable(Name="beta0shorta",Value=iprint(beta0shorta),Comment="");
variable(Name="beta1shorta",Value=iprint(beta1shorta),Comment="");
puts(What="Dummy2");
variable(Name="beta0dummy2a",Value=iprint(beta0dummy2a),Comment="");
variable(Name="beta1dummy2a",Value=iprint(beta1dummy2a),Comment="");
puts(What="Short");
variable(Name="beta0shortb",Value=iprint(beta0shortb),Comment="");
variable(Name="beta1shortb",Value=iprint(beta1shortb),Comment="");
puts(What="Dummy2");
variable(Name="beta0dummy2b",Value=iprint(beta0dummy2b),Comment="");
variable(Name="beta1dummy2b",Value=iprint(beta1dummy2b),Comment="");
puts(What="Short");
variable(Name="beta0shortc",Value=iprint(beta0shortc),Comment="");
variable(Name="beta1shortc",Value=iprint(beta1shortc),Comment="");
puts(What="Dummy2");
variable(Name="beta0dummy2c",Value=iprint(beta0dummy2c),Comment="");
variable(Name="beta1dummy2c",Value=iprint(beta1dummy2c),Comment="");

listcomment(Text="----------WaveguideStructure--------");
puts(What="--------------Stem--------------");
taper(pin=1,pout=2,offset[0,0,0],Start_Width=bigwidtha,End_Width=iprint(bigwidtha),Length=200,Offset=0,Angle_Out=0);
puts(What="-----------Outputs------------"); puts(What="Short");
SinBend(pin=2,pout=3,offset[0,bigwidtha/2-shorta/2,0],Width=iprint(shorta),Length=Ls,Offset=armoffset,Sections=20);
puts(What="Dummy2");
SinBend(pin=2,pout=4,offset[0,bigwidtha/2-shorta-dummy2a/2,0],Width=iprint(dummy2a),Length=Ls,Offset=-armoffset,Sections=20);

puts(What="--------------Stem--------------");
taper(pin=5,pout=6,offset[0,inoffset,0],Start_Width=bigwidthb,End_Width=iprint(bigwidthb),Length=200,Offset=0,Angle_Out=0);
puts(What="-----------Outputs------------"); puts(What="Short");
SinBend(pin=6,pout=7,offset[0,bigwidthb/2-shortb/2,0],Width=iprint(shortb),Length=Ls,Offset=armoffset,Sections=20);
puts(What="Dummy2");
SinBend(pin=6,pout=8,offset[0,bigwidthb/2-shortb-dummy2b/2,0],Width=iprint(dummy2b),Length=Ls,Offset=-armoffset,Sections=20);

puts(What="--------------Stem--------------");
```

```
taper(pin=9,pout=10,offset[0,2*inoffset,0],Start_Width=bigwidthc,End_Width=iprint(bigwidthc),Length=200,Offset=0,Angle_Out=0);
puts(What="Short");
SinBend(pin=10,pout=11,offset[0,bigwidthc/2-shortc/2,0],Width=iprint(shortc),Length=Ls,Offset=armoffset,Sections=20);
puts(What="Dummy2");
SinBend(pin=10,pout=12,offset[0,bigwidthc/2-shortc-dummy2c/2,0],Width=iprint(dummy2c),Length=Ls,Offset=-armoffset,Sections=20);


listcomment(Text="------------Input Source---------------");
fieldM(offset[1,0,0],Mode_No=2,Range=bigwidtha+2,Power=0.5,Phase=0,Tilt=0);
fieldM(offset[1,0,0],Mode_No=3,Range=bigwidtha+2,Power=0.5,Phase=0,Tilt=0);
fieldM(offset[1,inoffset,0],Mode_No=3,Range=bigwidthb+2,Power=1.7,Phase=0,Tilt=0);
fieldM(offset[1,2*inoffset,0],Mode_No=2,Range=bigwidthc+2,Power=1.7,Phase=0,Tilt=0);

puts(What="------------Mode Overlap---------------");
overlapM(pin=2,offset[+5,0,0],Label="bstem
1a",Mode_No=0,Range=bigwidtha  +2,Tilt=0);
overlapM(pin=3,offset[-5,0,0],Label="short
0a",Mode_No=0,Range=shorta+2,Tilt=0);
overlapM(pin=3,offset[-5,0,0],Label="short
1a",Mode_No=1,Range=shorta+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="dummy2
0a",Mode_No=0,Range=dummy2a+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="dummy2
1a",Mode_No=1,Range=dummy2a+2,Tilt=0);

overlapM(pin=6,offset[+5,0,0],Label="bstem
1a",Mode_No=1,Range=bigwidthb  +2,Tilt=0);
overlapM(pin=7,offset[-5,0,0],Label="short
0a",Mode_No=0,Range=shorta+2,Tilt=0);
overlapM(pin=7,offset[-5,0,0],Label="short
1a",Mode_No=1,Range=shorta+2,Tilt=0);
overlapM(pin=8,offset[-5,0,0],Label="dummy2
0a",Mode_No=0,Range=dummy2a+2,Tilt=0);
overlapM(pin=8,offset[-5,0,0],Label="dummy2
1a",Mode_No=1,Range=dummy2a+2,Tilt=0);

overlapM(pin=10,offset[+5,0,0],Label="bstem
1a",Mode_No=1,Range=bigwidthb  +2,Tilt=0);
overlapM(pin=11,offset[-5,0,0],Label="short
0a",Mode_No=0,Range=shorta+2,Tilt=0);
overlapM(pin=11,offset[-5,0,0],Label="short
1a",Mode_No=1,Range=shorta+2,Tilt=0);
overlapM(pin=12,offset[-5,0,0],Label="dummy2
0a",Mode_No=0,Range=dummy2a+2,Tilt=0);
overlapM(pin=12,offset[-5,0,0],Label="dummy2
1a",Mode_No=1,Range=dummy2a+2,Tilt=0);
```

## Reverse Y-junction

```
/* BBV written DEV file */ /* Contact us at bbv@bbv.nl */

dev_version 410;
MaskDescript(masksize=5,maskReferencePoint=[0,0]);

Material { MAT "Background" : STATE 1 : BACKGROUND : INDEX
"simDim()==3?1946:nback " : COLOR 255,255,255  : THERM "1","1" :
THOPT "1e-005","100000" : ELECT "1","1" : ELOPT "1e-006","1e-006"
: PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT "Waveguide" :
STATE 1 : INDEX "simDim()==3?19468:ncore " : COLOR 128,255,0  :
THERM "1","1" : THOPT "1e-005","100000" : ELECT "1","1" : ELOPT
"1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT
"start" : STATE 1 : INDEX "simDim()==3?1945568:nstart" : COLOR
255,88,0  : THERM "1","1" : THOPT "1e-005","100000" : ELECT
"1","1" : ELOPT "1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" :
GDSLAYER 0;
 };

layer("System",systemtype,view=off,fill=off,width=65,refPlaneY=0,overlayfactor=1,Emk[15240,0,0],ColorRGB(255,255,255))
{
 };
layer("Waveguide",view=on,fill=off,width=25,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(255,200,0))
{
  XY_CrossSection(Material="Waveguide",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="Waveguide",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=5));
 };
layer("start",view=on,fill=off,width=73,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(205,255,0))
{
  XY_CrossSection(Material="start",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="start",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=85));
 };

DefaultValue(Lambda=1.55,Width=65,ModeNum=0,Polarisation=TE,BpmOrder=2,viewmaskwindow=off,viewbpmwindow=on,viewmode="Normal",DiscretiseBpm=1e-020);
puts(What=""); listcomment(Text="-----------------Definition of
Device------------------"); deflayer(layer "Waveguide");
variable(Name="nback",Value=1.5,Comment="");
variable(Name="ncore",Value=nback+0.027,Comment="");
variable(Name="nstart",Value=ncore,Comment="");
listcomment(Text="---------------Stem----------------");
variable(Name="bigwidth",Value=12,Comment="");
listcomment(Text="------------Outputs--------------");
variable(Name="shorta",Value=bigwidth/2,Comment="");
variable(Name="dummy2a",Value=(bigwidth/2),Comment="");
variable(Name="shortb",Value=bigwidth/2,Comment="");
variable(Name="dummy2b",Value=(bigwidth/2),Comment="");
```

```
variable(Name="shortc",Value=bigwidth/2,Comment="");
variable(Name="dummy2c",Value=bigwidth/2,Comment="");
variable(Name="armoffset",Value=6,Comment="");
variable(Name="inoffset",Value=40,Comment="");
variable(Name="smode",Value=0,Comment="");
variable(Name="topphase",Value=180,Comment="");
listcomment(Text="---Output length-------");
variable(Name="Ls",Value=5000,Comment="");
listcomment(Text="--------Radius Factor--------");
variable(Name="rad",Value=0.0001,Comment="");
variable(Name="rad1",Value=0.1,Comment="");
puts(What="---------Beta Values---------");
variable(Name="beta0big",Value="modepropml('re','neff',0,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta1big",Value="modepropml('re','neff',1,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta2big",Value="modepropml('re','neff',2,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta3big",Value="modepropml('re','neff',3,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta4big",Value="modepropml('re','neff',4,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta5big",Value="modepropml('re','neff',5,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta0shorta",Value="modepropml('re','neff',0,5,nback,shorta,ncore,5,nback)",Comment="");
variable(Name="beta1shorta",Value="modepropml('re','neff',1,5,nback,shorta,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2a",Value="modepropml('re','neff',0,5,nback,dummy2a,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2a",Value="modepropml('re','neff',1,5,nback,dummy2a,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2a",Value="modepropml('re','neff',2,5,nback,dummy2a,ncore,5,nback)",Comment="");

variable(Name="beta0shortb",Value="modepropml('re','neff',0,5,nback,shortb,ncore,5,nback)",Comment="");
variable(Name="beta1shortb",Value="modepropml('re','neff',1,5,nback,shortb,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2b",Value="modepropml('re','neff',0,5,nback,dummy2b,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2b",Value="modepropml('re','neff',1,5,nback,dummy2b,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2b",Value="modepropml('re','neff',2,5,nback,dummy2b,ncore,5,nback)",Comment="");

variable(Name="beta0shortc",Value="modepropml('re','neff',0,5,nback,shortc,ncore,5,nback)",Comment="");
variable(Name="beta1shortc",Value="modepropml('re','neff',1,5,nback,shortc,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2c",Value="modepropml('re','neff',0,5,nback,dummy2c,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2c",Value="modepropml('re','neff',1,5,nback,dummy2c,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2c",Value="modepropml('re','neff',2,5,nback,dummy2c,ncore,5,nback)",Comment="");

puts(What="Bigstem");
variable(Name="beta1big",Value=iprint(beta1big),Comment="");
variable(Name="beta2big",Value=iprint(beta2big),Comment="");
variable(Name="beta3big",Value=iprint(beta3big),Comment="");
puts(What="Short");
variable(Name="beta0shorta",Value=iprint(beta0shorta),Comment="");
variable(Name="beta1shorta",Value=iprint(beta1shorta),Comment="");
puts(What="Dummy2");
variable(Name="beta0dummy2a",Value=iprint(beta0dummy2a),Comment="");
variable(Name="beta1dummy2a",Value=iprint(beta1dummy2a),Comment="");
puts(What="Short");
variable(Name="beta0shortb",Value=iprint(beta0shortb),Comment="");
variable(Name="beta1shortb",Value=iprint(beta1shortb),Comment="");
puts(What="Dummy2");
variable(Name="beta0dummy2b",Value=iprint(beta0dummy2b),Comment="");
variable(Name="beta1dummy2b",Value=iprint(beta1dummy2b),Comment="");
puts(What="Short");
variable(Name="beta0shortc",Value=iprint(beta0shortc),Comment="");
variable(Name="beta1shortc",Value=iprint(beta1shortc),Comment="");
puts(What="Dummy2");
variable(Name="beta0dummy2c",Value=iprint(beta0dummy2c),Comment="");
variable(Name="beta1dummy2c",Value=iprint(beta1dummy2c),Comment="");

listcomment(Text="----------WaveguideStructure--------");

puts(What="----------Inputs------------"); puts(What="Short");
SinBend(pin=7,offset[0,-bigwidth/2-2*armoffset,0],Width=iprint(shortb),Length=Ls,Offset=armoffset,Sections=20);
puts(What="Dummy2");
SinBend(pin=8,pout=6,offset[0,0,0],Width=iprint(dummy2b),Length=Ls,Offset=-armoffset,Sections=20);
puts(What="--------------Stem--------------");
taper(pin=6,pout=5,offset[0,-bigwidth/4,0],Start_Width=bigwidth,End_Width=iprint(bigwidth),Length=200,Offset=0,Angle_Out=0);

puts(What="----------Inputs------------"); puts(What="Short");
SinBend(pin=7,offset[0,inoffset-bigwidth/4-armoffset],Width=iprint(shortb),Length=Ls,Offset=armoffset,Sections=20);
puts(What="Dummy2");
SinBend(pin=8,pout=6,offset[0,inoffset+bigwidth/4+armoffset,0],Width=iprint(dummy2b),Length=Ls,Offset=-armoffset,Sections=20);
puts(What="--------------Stem--------------");
taper(pin=6,pout=5,offset[0,-bigwidth/4,0],Start_Width=bigwidth,End_Width=iprint(bigwidth),Length=200,Offset=0,Angle_Out=0);

puts(What="----------Inputs------------"); puts(What="Short");
SinBend(pin=11,offset[0,2*inoffset-bigwidth/4-armoffset],Width=iprint(shortb),Length=Ls,Offset=armoffset,Sections=20);
puts(What="Dummy2");
SinBend(pin=12,pout=10,offset[0,2*inoffset+bigwidth/4+armoffset,0],Width=iprint(dummy2b),Length=Ls,Offset=-armoffset,Sections=20);
puts(What="--------------Stem--------------");
taper(pin=10,pout=9,offset[0,-bigwidth/4,0],Start_Width=bigwidth,End_Width=iprint(bigwidth),Length=200,Offset=0,Angle_Out=0);

puts(What="----------Inputs------------"); puts(What="Short");
SinBend(pin=13,offset[0,3*inoffset-bigwidth/4-armoffset],Width=iprint(shortb),Length=Ls,Offset=armoffset,Sections=20);
puts(What="Dummy2");
SinBend(pin=14,pout=15,offset[0,3*inoffset+bigwidth/4+armoffset,0],Width=iprint(dummy2b),Length=Ls,Offset=-armoffset,Sections=20);
puts(What="--------------Stem--------------");
taper(pin=15,pout=16,offset[0,-bigwidth/4,0],Start_Width=bigwidth,End_Width=iprint(bigwidth),Length=200,Offset=0,Angle_Out=0);



listcomment(Text="-----------Input Source---------------");
fieldM(offset[1,0,0],Mode_No=1,Range=bigwidth/2+2,Power=1,Phase=0,Tilt=0);
fieldM(offset[1,inoffset-bigwidth/4-armoffset,0],Mode_No=1,Range=bigwidth/2+2,Power=1,Phase=0,Tilt=0);
fieldM(offset[1,inoffset+bigwidth/4+armoffset,0],Mode_No=1,Range=bigwidth/2+2,Power=1,Phase=0,Tilt=0);
fieldM(offset[1,2*inoffset-bigwidth/4-armoffset,0],Mode_No=1,Range=bigwidth/2+2,Power=0.5,Phase=0,Tilt=0);
```

```
fieldM(offset[1,2*inoffset+bigwidth/4+armoffset,0],Mode_No=1,Range=bigwidth/2+2,Power=0.5,Phase=90,Tilt=0);
fieldM(offset[1,3*inoffset-bigwidth/4-armoffset,0],Mode_No=1,Range=bigwidth/2+2,Power=1,Phase=0,Tilt=0);
fieldM(offset[1,3*inoffset+bigwidth/4+armoffset,0],Mode_No=1,Range=bigwidth/2+2,Power=1,Phase=180,Tilt=0);


puts(What="------------Mode Overlap--------------");
overlapM(pin=2,offset[+5,0,0],Label="bstem
1a",Mode_No=smode,Range=bigwidth  +2,Tilt=0);
overlapM(pin=3,offset[-5,0,0],Label="short
0a",Mode_No=0,Range=shorta+2,Tilt=0);
overlapM(pin=3,offset[-5,0,0],Label="short
1a",Mode_No=1,Range=shorta+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="dummy2
0a",Mode_No=0,Range=dummy2a+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="dummy2
1a",Mode_No=1,Range=dummy2a+2,Tilt=0);

overlapM(pin=5,offset[-5,0,0],Label="bstem
2b",Mode_No=0,Range=bigwidth  +2,Tilt=0);
overlapM(pin=5,offset[-5,0,0],Label="bstem
2b",Mode_No=1,Range=bigwidth  +2,Tilt=0);
overlapM(pin=7,offset[+5,0,0],Label="short
0b",Mode_No=0,Range=shortb+2,Tilt=0);
overlapM(pin=7,offset[+5,0,0],Label="short
1b",Mode_No=1,Range=shortb+2,Tilt=0);
overlapM(pin=8,offset[+5,0,0],Label="dummy2
0b",Mode_No=0,Range=dummy2b+2,Tilt=0);
overlapM(pin=8,offset[+5,0,0],Label="dummy2
1b",Mode_No=1,Range=dummy2b+2,Tilt=0);

overlapM(pin=9,offset[-5,0,0],Label="bstem
2b",Mode_No=0,Range=bigwidth  +2,Tilt=0);
overlapM(pin=9,offset[-5,0,0],Label="bstem
2b",Mode_No=1,Range=bigwidth  +2,Tilt=0);
overlapM(pin=11,offset[+5,0,0],Label="short
0b",Mode_No=0,Range=shortb+2,Tilt=0);
overlapM(pin=11,offset[+5,0,0],Label="short
1b",Mode_No=1,Range=shortb+2,Tilt=0);
overlapM(pin=12,offset[+5,0,0],Label="dummy2
0b",Mode_No=0,Range=dummy2b+2,Tilt=0);
overlapM(pin=12,offset[+5,0,0],Label="dummy2
1b",Mode_No=1,Range=dummy2b+2,Tilt=0);

overlapM(pin=16,offset[-5,0,0],Label="bstem3c",Mode_No=0,Range=bigwidth
+2,Tilt=0);
overlapM(pin=16,offset[-5,0,0],Label="bstem3c",Mode_No=1,Range=bigwidth
+2,Tilt=0); overlapM(pin=13,offset[+5,0,0],Label="short
0c",Mode_No=0,Range=shortc+2,Tilt=0);
overlapM(pin=13,offset[+5,0,0],Label="short
1c",Mode_No=1,Range=shortc+2,Tilt=0);
overlapM(pin=14,offset[+5,0,0],Label="dummy2
0c",Mode_No=0,Range=dummy2c+2,Tilt=0);
overlapM(pin=14,offset[+5,0,0],Label="dummy2
1c",Mode_No=1,Range=dummy2c+2,Tilt=0);
```

## 8.3.2 Device

The BBV DEV file for the 5-way device is provided. The 2, 3, and 4 way devices are simplifications of the 5-way device.

## Demultiplexing

```
/* BBV written DEV file */ /* Contact us at bbv@bbv.nl */

dev_version 410;
MaskDescript(masksize=5,maskReferencePoint=[0,0]);

Material { MAT "Background" : STATE 1 : BACKGROUND : INDEX
"simDim()==3?1946:nback " : COLOR 255,255,255  : THERM "1","1" :
THOPT "1e-005","100000" : ELECT "1","1" : ELOPT "1e-006","1e-006"
: PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT "Waveguide" :
STATE 1 : INDEX "simDim()==3?19468:ncore " : COLOR 128,255,0  :
THERM "1","1" : THOPT "1e-005","100000" : ELECT "1","1" : ELOPT
"1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT
"start" : STATE 1 : INDEX "simDim()==3?1945568:nstart" : COLOR
255,88,0  : THERM "1","1" : THOPT "1e-005","100000" : ELECT
"1","1" : ELOPT "1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" :
GDSLAYER 0;
 };

layer("System",systemtype,view=off,fill=off,width=65,refPlaneY=0,overlayfactor=1,Emk[15240,0,0],ColorRGB(255,255,255))
{
 };
layer("Waveguide",view=on,fill=off,width=25,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(255,200,0))
{
  XY_CrossSection(Material="Waveguide",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="Waveguide",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=5));
 };
layer("start",view=on,fill=off,width=73,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(205,255,0))
{
  XY_CrossSection(Material="start",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="start",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=85));
 };
```

```
DefaultValue(Lambda=0.9,Width=65,ModeNum=0,Polarisation=TE,BpmOrder=2,viewmaskwindow=off,viewbpmwindow=on,viewmode="Normal",DiscretiseBpm=1e-020);
listcomment(Text="-----------------Definition of
Device------------------"); deflayer(layer "Waveguide");
variable(Name="nback",Value=1.5,Comment="");
variable(Name="ncore",Value=nback+0.027,Comment="");
variable(Name="nstart",Value=ncore,Comment="");
listcomment(Text="---------------input----------------");
variable(Name="tolerance",Value=0,Comment="");
variable(Name="inwidth",Value=0.75+ 0.75*tolerance,Comment="");
listcomment(Text="---------------Stem----------------");
variable(Name="recwidth",Value=7.2,Comment="");
variable(Name="bigwidth",Value=25,Comment="");
listcomment(Text="------------Outputs--------------");
variable(Name="short",Value=3.35,Comment="");
variable(Name="mid",Value=4,Comment="");
variable(Name="Long",Value=4.8,Comment="");
variable(Name="dummy1",Value=7,Comment="");
variable(Name="dummy2",Value=(bigwidth-short-mid-dummy1-Long),Comment="");
listcomment(Text="----input  length-------");
variable(Name="Ls1",Value=1500,Comment="");
listcomment(Text="----output length-------");
variable(Name="Ls",Value=6500,Comment="");
listcomment(Text="----output offset-------");
variable(Name="outoffset",Value=9,Comment="");
listcomment(Text="--------Radius Factor--------");
variable(Name="rad",Value=0.1,Comment="");
variable(Name="rad1",Value=0.1,Comment="");
puts(What="---------Beta Values---------");
variable(Name="beta0input",Value="modepropml('re','neff',0,5,nback,inwidth,ncore,5,nback)",Comment="");
variable(Name="beta1stem",Value="modepropml('re','neff',1,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta2stem",Value="modepropml('re','neff',2,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta3stem",Value="modepropml('re','neff',3,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta0big",Value="modepropml('re','neff',0,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta1big",Value="modepropml('re','neff',1,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta2big",Value="modepropml('re','neff',2,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta3big",Value="modepropml('re','neff',3,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta4big",Value="modepropml('re','neff',4,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta5big",Value="modepropml('re','neff',5,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta0short",Value="modepropml('re','neff',0,5,nback,short,ncore,5,nback)",Comment="");
variable(Name="beta1short",Value="modepropml('re','neff',1,5,nback,short,ncore,5,nback)",Comment="");
variable(Name="beta0mid",Value="modepropml('re','neff',0,5,nback,mid,ncore,5,nback)",Comment="");
variable(Name="beta1mid",Value="modepropml('re','neff',1,5,nback,mid,ncore,5,nback)",Comment="");
variable(Name="beta0Long",Value="modepropml('re','neff',0,5,nback,Long,ncore,5,nback)",Comment="");
variable(Name="beta1Long",Value="modepropml('re','neff',1,5,nback,Long,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2",Value="modepropml('re','neff',0,5,nback,dummy2,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2",Value="modepropml('re','neff',1,5,nback,dummy2,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2",Value="modepropml('re','neff',2,5,nback,dummy2,ncore,5,nback)",Comment="");
puts(What="input");
variable(Name="beta0input",Value=iprint(beta0input),Comment="");
puts(What="stem");
variable(Name="beta1stem",Value=iprint(beta1stem),Comment="");
variable(Name="beta2stem",Value=iprint(beta2stem),Comment="");
variable(Name="beta3stem",Value=iprint(beta3stem),Comment="");
puts(What="Bigstem");
variable(Name="beta1big",Value=iprint(beta1big),Comment="");
variable(Name="beta2big",Value=iprint(beta2big),Comment="");
variable(Name="beta3big",Value=iprint(beta3big),Comment="");
puts(What="Short");
variable(Name="beta0short",Value=iprint(beta0short),Comment="");
puts(What="Mid");
variable(Name="beta0mid",Value=iprint(beta0mid),Comment="");
puts(What="Long");
variable(Name="beta0Long",Value=iprint(beta0Long),Comment="");
puts(What="Dummy2");
variable(Name="beta1dummy2",Value=iprint(beta1dummy2),Comment="");
variable(Name="beta2dummy2",Value=iprint(beta2dummy2),Comment="");
listcomment(Text="----------WaveguideStructure--------");
puts(What="-------------inputs--------------");
puts(What="----Width----"); SinBend(layer
"Waveguide",pin=1,pout=12,Width=iprint(inwidth),Length=Ls1,Offset=0,Sections=4);
SinBend(layer
"Waveguide",pout=11,offset[0,-4-recwidth/2+(recwidth-inwidth)/4,0],Width=iprint((recwidth-inwidth)/2),Length=Ls1,Offset=4,Sections=20);
SinBend(layer
"Waveguide",pout=13,offset[0,-6-recwidth/2-(recwidth-inwidth)/4,0],Width=iprint((recwidth-inwidth)/2),Length=Ls1,Offset=6,Sections=20);
puts(What="--------------Stem--------------");
puts(What="----Width----");
taper(pin=12,pout=14,offset[0,1*inwidth/2-recwidth/2,0],Start_Width=iprint(recwidth),End_Width=recwidth,Length=400,Offset=0,Angle_Out=0);
taper(pin=14,pout=22,Start_Width=recwidth,End_Width=iprint(bigwidth),Length=100*abs(bigwidth-recwidth),Offset=0,Angle_Out=0);
taper(pin=22,pout=2,Start_Width=bigwidth,End_Width=bigwidth,Length=200,Offset=0,Angle_Out=0);
puts(What="-----------Outputs------------");
puts(What="----Width----"); puts(What="Dummy1"); SinBend(layer
"Waveguide",pin=2,pout=3,offset[0,bigwidth/2-dummy1/2,0],Width=iprint(dummy1),Length=Ls,Offset=2*outoffset,Sections=30);
puts(What="Short"); SinBend(layer
"Waveguide",pin=2,pout=4,offset[0,bigwidth/2-dummy1-short/2,0],Width=iprint(short),Length=Ls,Offset=outoffset,Sections=30);
puts(What="Mid");
taper(pin=2,pout=5,offset[0,bigwidth/2-dummy1-short-
mid/2,0],Start_Width=mid,End_Width=iprint(mid),Length=Ls,Offset=0,Angle_Out=0);
puts(What="Long"); SinBend(layer
"Waveguide",pin=2,pout=6,offset[0,bigwidth/2-dummy1-short-mid-Long/2,0],Width=iprint(Long),Length=Ls,Offset=-outoffset,Sections=30);
puts(What="Dummy2"); SinBend(layer
"Waveguide",pin=2,pout=7,offset[0,bigwidth/2-dummy1-short-mid-Long-dummy2/2,0],Width=iprint(dummy2),Length=Ls,Offset=-2*outoffset,Sections=40);
listcomment(Text="------------Input Source---------------");
fieldM(offset[1,0,0],Mode_No=0,Range=inwidth+2,Power=1,Phase=0,Tilt=0);
listcomment(Text="------------Mode Overlap---------------");
overlapM(pin=14,offset[-5,0,0],Label="stem
1",Mode_No=1,Range=recwidth  +2,Tilt=0);
```

```
overlapM(pin=14,offset[-5,0,0],Label="stem
2",Mode_No=2,Range=recwidth  +2,Tilt=0);
overlapM(pin=14,offset[-5,0,0],Label="stem3",Mode_No=3,Range=recwidth
+2,Tilt=0); overlapM(pin=2,offset[-5,0,0],Label="bstem
1",Mode_No=1,Range=bigwidth  +2,Tilt=0);
overlapM(pin=2,offset[-5,0,0],Label="bstem
2",Mode_No=2,Range=bigwidth  +2,Tilt=0);
overlapM(pin=2,offset[-5,0,0],Label="bstem3",Mode_No=3,Range=bigwidth
+2,Tilt=0);
overlapM(pin=2,offset[-5,0,0],Label="bstem4",Mode_No=4,Range=bigwidth
+2,Tilt=0); overlapM(pin=3,offset[-5,0,0],Label="dummy1
0",Mode_No=0,Range=dummy1+2,Tilt=0);
overlapM(pin=3,offset[-5,0,0],Label="dummy1
1",Mode_No=1,Range=dummy1+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="short
0",Mode_No=0,Range=short+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="short
1",Mode_No=1,Range=short+2,Tilt=0);
overlapM(pin=5,offset[-5,0,0],Label="Mid 0",Mode_No=0,Range=mid
+2,Tilt=0); overlapM(pin=5,offset[-5,0,0],Label="Mid
1",Mode_No=1,Range=mid+2,Tilt=0);
overlapM(pin=6,offset[-5,0,0],Label="Long 0",Mode_No=0,Range=Long
+2,Tilt=0); overlapM(pin=6,offset[-5,0,0],Label="Long
1",Mode_No=1,Range=Long+2,Tilt=0);
overlapM(pin=7,offset[-5,0,0],Label="dummy2
0",Mode_No=0,Range=dummy2+2,Tilt=0);
overlapM(pin=7,offset[-5,0,0],Label="dummy2
1",Mode_No=1,Range=dummy2+2,Tilt=0);
```

## Multiplexing

```
/* BBV written DEV file */ /* Contact us at bbv@bbv.nl */

dev_version 410;
MaskDescript(masksize=5,maskReferencePoint=[0,0]);

Material { MAT "Background" : STATE 1 : BACKGROUND : INDEX
"simDim()==3?1946:nback " : COLOR 255,255,255  : THERM "1","1" :
THOPT "1e-005","100000" : ELECT "1","1" : ELOPT "1e-006","1e-006"
: PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT "Waveguide" :
STATE 1 : INDEX "simDim()==3?19468:ncore " : COLOR 128,255,0  :
THERM "1","1" : THOPT "1e-005","100000" : ELECT "1","1" : ELOPT
"1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT
"start" : STATE 1 : INDEX "simDim()==3?1945568:nstart" : COLOR
255,88,0  : THERM "1","1" : THOPT "1e-005","100000" : ELECT
"1","1" : ELOPT "1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" :
GDSLAYER 0;
 };

layer("System",systemtype,view=off,fill=off,width=65,refPlaneY=0,overlayfactor=1,Emk[15240,0,0],ColorRGB(255,255,255))
{
 };
layer("Waveguide",view=on,fill=off,width=25,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(255,200,0))
{
  XY_CrossSection(Material="Waveguide",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="Waveguide",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=5));
 };
layer("start",view=on,fill=off,width=73,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(205,255,0))
{
  XY_CrossSection(Material="start",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="start",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=85));
 };

DefaultValue(Lambda=1.3,Width=65,ModeNum=0,Polarisation=TE,BpmOrder=2,viewmaskwindow=off,viewbpmwindow=on,viewmode="Normal",DiscretiseBpm=1e-020);
listcomment(Text="-----------------Definition of
Device------------------"); deflayer(layer "Waveguide");
variable(Name="nback",Value=1.5,Comment="");
variable(Name="ncore",Value=nback+0.027,Comment="");
variable(Name="nstart",Value=ncore,Comment="");
listcomment(Text="---------------input----------------");
variable(Name="inwidth",Value=0.75,Comment="");
variable(Name="phantom",Value=3.2,Comment="");
listcomment(Text="---------------Stem----------------");
variable(Name="recwidth",Value=7.2,Comment="");
variable(Name="bigwidth",Value=25,Comment="");
listcomment(Text="------------Outputs--------------");
variable(Name="dummy1",Value=7,Comment="");
variable(Name="short",Value=3.35,Comment="");
variable(Name="mid",Value=4,Comment="");
variable(Name="Long",Value=4.8,Comment="");
variable(Name="dummy2",Value=(bigwidth-dummy1-short-mid-Long),Comment="");
listcomment(Text="----input  length-------");
variable(Name="Ls1",Value=1500,Comment="");
listcomment(Text="----input  offset-------");
variable(Name="inoffset",Value=9,Comment="");
listcomment(Text="----output length-------");
variable(Name="Ls",Value=6500,Comment="");
listcomment(Text="--------Radius Factor--------");
variable(Name="rad",Value=0.1,Comment="");
variable(Name="rad1",Value=0.1,Comment="");
puts(What="---------Beta Values---------");
variable(Name="beta0input",Value="modepropml('re','neff',0,5,nback,inwidth,ncore,5,nback)",Comment="");
variable(Name="beta1stem",Value="modepropml('re','neff',1,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta2stem",Value="modepropml('re','neff',2,5,nback,recwidth,ncore,5,nback)",Comment="");
```

```
variable(Name="beta3stem",Value="modepropml('re','neff',3,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta0big",Value="modepropml('re','neff',0,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta1big",Value="modepropml('re','neff',1,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta2big",Value="modepropml('re','neff',2,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta3big",Value="modepropml('re','neff',3,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta4big",Value="modepropml('re','neff',4,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta5big",Value="modepropml('re','neff',5,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta0short",Value="modepropml('re','neff',0,5,nback,short,ncore,5,nback)",Comment="");
variable(Name="beta1short",Value="modepropml('re','neff',1,5,nback,short,ncore,5,nback)",Comment="");
variable(Name="beta0mid",Value="modepropml('re','neff',0,5,nback,mid,ncore,5,nback)",Comment="");
variable(Name="beta1mid",Value="modepropml('re','neff',1,5,nback,mid,ncore,5,nback)",Comment="");
variable(Name="beta0Long",Value="modepropml('re','neff',0,5,nback,Long,ncore,5,nback)",Comment="");
variable(Name="beta1Long",Value="modepropml('re','neff',1,5,nback,Long,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2",Value="modepropml('re','neff',0,5,nback,dummy2,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2",Value="modepropml('re','neff',1,5,nback,dummy2,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2",Value="modepropml('re','neff',2,5,nback,dummy2,ncore,5,nback)",Comment="");
puts(What="input");
variable(Name="beta0input",Value=iprint(beta0input),Comment="");
puts(What="stem");
variable(Name="beta1stem",Value=iprint(beta1stem),Comment="");
variable(Name="beta2stem",Value=iprint(beta2stem),Comment="");
variable(Name="beta3stem",Value=iprint(beta3stem),Comment="");
puts(What="Bigstem");
variable(Name="beta1big",Value=iprint(beta1big),Comment="");
variable(Name="beta2big",Value=iprint(beta2big),Comment="");
variable(Name="beta3big",Value=iprint(beta3big),Comment="");
puts(What="Short");
variable(Name="beta0short",Value=iprint(beta0short),Comment="");
puts(What="Mid");
variable(Name="beta0mid",Value=iprint(beta0mid),Comment="");
puts(What="Long");
variable(Name="beta0Long",Value=iprint(beta0Long),Comment="");
puts(What="Dummy2");
variable(Name="beta1dummy2",Value=iprint(beta1dummy2),Comment="");
variable(Name="beta2dummy2",Value=iprint(beta2dummy2),Comment="");
listcomment(Text="----------WaveguideStructure--------");
puts(What="-------------inputs--------------");
puts(What="----Width----"); puts(What="Dummy1"); SinBend(layer
"Waveguide",pin=3,pout=20,offset[0,2*inoffset,0],Width=iprint(dummy1),Length=Ls,Offset=-2*inoffset+mid/2+short+dummy1/2,Sections=20);

puts(What="Short"); SinBend(layer
"Waveguide",pin=4,pout=20,offset[0,inoffset,0],Width=iprint(short),Length=Ls,Offset=-inoffset+mid/2+short/2,Sections=20);
puts(What="Mid");
taper(pin=5,pout=22,Start_Width=mid,End_Width=iprint(mid),Length=Ls,Offset=0,Angle_Out=0);
puts(What="Long"); SinBend(layer
"Waveguide",pin=6,pout=21,offset[0,-inoffset,0],Width=iprint(Long),Length=Ls,Offset=inoffset-mid/2-Long/2,Sections=20);
puts(What="Dummy2"); SinBend(layer
"Waveguide",pin=7,pout=23,offset[0,-2*inoffset,0],Width=iprint(dummy2),Length=Ls,Offset=2*inoffset-mid/2-Long-dummy2/2,Sections=20);
puts(What="--------------Stem--------------");
puts(What="----Width----");
taper(pin=22,pout=2,offset[0,bigwidth/2-mid/2-Long-dummy2,0],Start_Width=bigwidth,End_Width=bigwidth,Length=200,Offset=0,Angle_Out=0);
taper(pin=2,pout=14,Start_Width=iprint(bigwidth),End_Width=recwidth,Length=100*abs(bigwidth-recwidth),Offset=0,Angle_Out=0);
taper(pin=14,pout=12,Start_Width=iprint(recwidth),End_Width=recwidth,Length=400,Offset=0,Angle_Out=0);
puts(What="-----------Outputs------------");
puts(What="----Width----"); SinBend(layer
"Waveguide",pin=12,pout=1,offset[0,inwidth/2-recwidth/2,0],Width=iprint(inwidth),Length=Ls1,Offset=0,Sections=4);
SinBend(layer
"Waveguide",pin=12,pout=10,offset[0,-recwidth/2+inwidth+phantom/2,0],Width=iprint(phantom),Length=Ls1,Offset=4,Sections=20);
SinBend(layer
"Waveguide",pin=12,pout=11,offset[0,recwidth/2-(recwidth-inwidth-phantom)/2,0],Width=iprint(recwidth-inwidth-phantom),Length=Ls1,Offset=8,Sections=20)
listcomment(Text="------------Input Source---------------");
fieldM(offset[1,0,0],Mode_No=0,Range=mid+2,Power=1,Phase=0,Tilt=0);
listcomment(Text="------------Mode Overlap---------------");
overlapM(pin=14,offset[5,0,0],Label="stem
1",Mode_No=1,Range=recwidth  +2,Tilt=0);
overlapM(pin=14,offset[5,0,0],Label="stem
2",Mode_No=2,Range=recwidth  +2,Tilt=0);
overlapM(pin=14,offset[5,0,0],Label="stem3",Mode_No=3,Range=recwidth
+2,Tilt=0); overlapM(pin=2,offset[5,0,0],Label="bstem
1",Mode_No=1,Range=bigwidth  +2,Tilt=0);
overlapM(pin=2,offset[5,0,0],Label="bstem
2",Mode_No=2,Range=bigwidth  +2,Tilt=0);
overlapM(pin=2,offset[5,0,0],Label="bstem3",Mode_No=3,Range=bigwidth
+2,Tilt=0);
overlapM(pin=2,offset[5,0,0],Label="bstem4",Mode_No=4,Range=bigwidth
+2,Tilt=0); overlapM(pin=4,offset[5,0,0],Label="short
0",Mode_No=0,Range=short+2,Tilt=0);
overlapM(pin=4,offset[5,0,0],Label="short
1",Mode_No=1,Range=short+2,Tilt=0);
overlapM(pin=5,offset[5,0,0],Label="Mid 0",Mode_No=0,Range=mid
+2,Tilt=0); overlapM(pin=5,offset[5,0,0],Label="Mid
1",Mode_No=1,Range=mid+2,Tilt=0);
overlapM(pin=5,offset[5,0,0],Label="Long 0",Mode_No=0,Range=Long
+2,Tilt=0); overlapM(pin=5,offset[5,0,0],Label="Long
1",Mode_No=1,Range=Long+2,Tilt=0);
overlapM(pin=7,offset[5,0,0],Label="dummy2
0",Mode_No=0,Range=dummy2+2,Tilt=0);
overlapM(pin=7,offset[5,0,0],Label="dummy2
1",Mode_No=1,Range=dummy2+2,Tilt=0);
overlapM(pin=10,offset[-5,0,0],Label="phantom1
0",Mode_No=0,Range=phantom +2,Tilt=0);
overlapM(pin=10,offset[-5,0,0],Label="phantom1
1",Mode_No=1,Range=phantom +2,Tilt=0);
overlapM(pin=11,offset[-5,0,0],Label="phantom2
0",Mode_No=0,Range=recwidth-inwidth-phantom +2,Tilt=0);
overlapM(pin=11,offset[-5,0,0],Label="phantom2
```

```
1",Mode_No=1,Range=recwidth-inwidth-phantom +2,Tilt=0);
overlapM(pin=1,offset[-5,0,0],Label="Mux Output
0",Mode_No=0,Range=inwidth +2,Tilt=0);
overlapM(pin=1,offset[-5,0,0],Label="Mux Output
1",Mode_No=1,Range=inwidth+2,Tilt=0);
```

### 8.3.3 Adiabatic Taper simulation

```
/* BBV written DEV file */ /* Contact us at bbv@bbv.nl */

dev_version 410;
MaskDescript(masksize=5,maskReferencePoint=[0,0]);

Material { MAT "Background" : STATE 1 : BACKGROUND : INDEX
"simDim()==3?1946:nback " : COLOR 255,255,255  : THERM "1","1" :
THOPT "1e-005","100000" : ELECT "1","1" : ELOPT "1e-006","1e-006"
: PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT "Waveguide" :
STATE 1 : INDEX "simDim()==3?19468:ncore " : COLOR 128,255,0  :
THERM "1","1" : THOPT "1e-005","100000" : ELECT "1","1" : ELOPT
"1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT
"start" : STATE 1 : INDEX "simDim()==3?1945568:nstart" : COLOR
255,88,0  : THERM "1","1" : THOPT "1e-005","100000" : ELECT
"1","1" : ELOPT "1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" :
GDSLAYER 0;
 };

layer("System",systemtype,view=off,fill=off,width=65,refPlaneY=0,overlayfactor=1,Emk[15240,0,0],ColorRGB(255,255,255))
{
 };
layer("Waveguide",view=on,fill=off,width=25,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(255,200,0))
{
  XY_CrossSection(Material="Waveguide",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="Waveguide",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=5));
 };
layer("start",view=on,fill=off,width=73,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(205,255,0))
{
  XY_CrossSection(Material="start",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="start",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=85));
 };

DefaultValue(Lambda=1,Width=65,ModeNum=0,Polarisation=TE,BpmOrder=2,viewmaskwindow=off,viewbpmwindow=on,viewmode="Normal",DiscretiseBpm=1e-020);
listcomment(Text="-----------------Definition of
Device------------------"); deflayer(layer "Waveguide");
variable(Name="nback",Value=1.5,Comment="");
variable(Name="ncore",Value=nback+0.027,Comment="");
variable(Name="nstart",Value=ncore,Comment="");
listcomment(Text="---------------test----------------");
variable(Name="test",Value=3.95,Comment="");

listcomment(Text="---------------Stem----------------");
variable(Name="recwidth",Value=7.2,Comment="");
variable(Name="bigwidth",Value=20,Comment="");
variable(Name="gradient",Value=0.005,Comment="");

listcomment(Text="----input  length-------");
variable(Name="Ls1",Value=abs(recwidth-bigwidth)/gradient,Comment="");
variable(Name="smode",Value=2,Comment="");


variable(Name="beta1stem",Value="modepropml('re','neff',1,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta2stem",Value="modepropml('re','neff',2,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta3stem",Value="modepropml('re','neff',3,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta4stem",Value="modepropml('re','neff',4,5,nback,recwidth,ncore,5,nback)",Comment="");
variable(Name="beta0big",Value="modepropml('re','neff',0,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta1big",Value="modepropml('re','neff',1,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta2big",Value="modepropml('re','neff',2,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta3big",Value="modepropml('re','neff',3,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta4big",Value="modepropml('re','neff',4,5,nback,bigwidth,ncore,5,nback)",Comment="");

puts(What="stem");
variable(Name="beta1stem",Value=iprint(beta1stem),Comment="");
variable(Name="beta2stem",Value=iprint(beta2stem),Comment="");
variable(Name="beta3stem",Value=iprint(beta3stem),Comment="");
variable(Name="beta4stem",Value=iprint(beta4stem),Comment="");
puts(What="Bigstem");
variable(Name="beta1big",Value=iprint(beta1big),Comment="");
variable(Name="beta2big",Value=iprint(beta2big),Comment="");
variable(Name="beta3big",Value=iprint(beta3big),Comment="");
variable(Name="beta4big",Value=iprint(beta4big),Comment="");
listcomment(Text="----------WaveguideStructure--------");
puts(What="-------------inputs-------------");
puts(What="----Width----"); SinBend(layer
"Waveguide",pin=1,pout=2,Width=iprint(recwidth),Length=100,Offset=0,Sections=1);
puts(What="--------------Stem--------------");
puts(What="----Width----");
taper(pin=2,pout=3,offset[0,0,0],Start_Width=iprint(recwidth),End_Width=recwidth,Length=400,Offset=0,Angle_Out=0);
taper(pin=3,pout=4,Start_Width=recwidth,End_Width=iprint(bigwidth),Length=Ls1,Offset=0,Angle_Out=0);
taper(pin=4,pout=5,Start_Width=bigwidth,End_Width=bigwidth,Length=500,Offset=0,Angle_Out=0);
listcomment(Text="----------WaveguideStructure--------");
puts(What="-------------inputs-------------");
puts(What="----Width----"); SinBend(layer
"Waveguide",pin=6,pout=7,offset[0,30,0],Width=iprint(recwidth),Length=100,Offset=0,Sections=1);
puts(What="--------------Stem--------------");
puts(What="----Width----");
```

```
taper(pin=7,pout=8,offset[0,0,0],Start_Width=iprint(recwidth),End_Width=recwidth,Length=400,Offset=0,Angle_Out=0);
taper(pin=8,pout=9,Start_Width=recwidth,End_Width=iprint(bigwidth),Length=Ls1,Offset=0,Angle_Out=0);
taper(pin=9,pout=10,Start_Width=bigwidth,End_Width=bigwidth,Length=500,Offset=0,Angle_Out=0);
listcomment(Text="----------WaveguideStructure--------");
puts(What="-------------inputs--------------");
puts(What="----Width----"); SinBend(layer
"Waveguide",pin=11,pout=12,offset[0,60,0],Width=iprint(recwidth),Length=100,Offset=0,Sections=1);
puts(What="--------------Stem--------------");
puts(What="----Width----");
taper(pin=12,pout=13,offset[0,0,0],Start_Width=iprint(recwidth),End_Width=recwidth,Length=400,Offset=0,Angle_Out=0);
taper(pin=13,pout=14,Start_Width=recwidth,End_Width=iprint(bigwidth),Length=Ls1,Offset=0,Angle_Out=0);
taper(pin=14,pout=15,Start_Width=bigwidth,End_Width=bigwidth,Length=500,Offset=0,Angle_Out=0);
listcomment(Text="------------Input Source---------------");
fieldM(offset[1,0,0],Mode_No=1,Range=recwidth+2,Power=1,Phase=0,Tilt=0);
fieldM(offset[1,30,0],Mode_No=2,Range=recwidth+2,Power=1,Phase=0,Tilt=0);
fieldM(offset[1,60,0],Mode_No=3,Range=recwidth+2,Power=1,Phase=0,Tilt=0);
listcomment(Text="------------Mode Overlap---------------");

overlapM(pin=3,offset[-5,0,0],Label="stem
1a",Mode_No=1,Range=recwidth  +2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="bstem
1a",Mode_No=1,Range=bigwidth  +2,Tilt=0);

overlapM(pin=8,offset[-5,0,0],Label="stem
2b",Mode_No=2,Range=recwidth  +2,Tilt=0);
overlapM(pin=9,offset[-5,0,0],Label="bstem
2b",Mode_No=2,Range=bigwidth  +2,Tilt=0);

overlapM(pin=13,offset[-5,0,0],Label="stem
3c",Mode_No=3,Range=recwidth  +2,Tilt=0);
overlapM(pin=14,offset[-5,0,0],Label="bstem
3c",Mode_No=3,Range=bigwidth   +2,Tilt=0);
```

### 8.3.4 Adiabatic Bend Length

```
/* BBV written DEV file */ /* Contact us at bbv@bbv.nl */

dev_version 410;
MaskDescript(masksize=5,maskReferencePoint=[0,0]);

Material { MAT "Background" : STATE 1 : BACKGROUND : INDEX
"simDim()==3?1946:nback " : COLOR 255,255,255  : THERM "1","1" :
THOPT "1e-005","100000" : ELECT "1","1" : ELOPT "1e-006","1e-006"
: PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT "Waveguide" :
STATE 1 : INDEX "simDim()==3?19468:ncore " : COLOR 128,255,0  :
THERM "1","1" : THOPT "1e-005","100000" : ELECT "1","1" : ELOPT
"1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" : GDSLAYER 0; MAT
"start" : STATE 1 : INDEX "simDim()==3?1945568:nstart" : COLOR
255,88,0  : THERM "1","1" : THOPT "1e-005","100000" : ELECT
"1","1" : ELOPT "1e-006","1e-006" : PUMP_INDEX "1" : DOPING "0" :
GDSLAYER 0;
 };

layer("System",systemtype,view=off,fill=off,width=65,refPlaneY=0,overlayfactor=1,Emk[15240,0,0],ColorRGB(255,255,255))
{
 };
layer("Waveguide",view=on,fill=off,width=25,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(255,200,0))
{
  XY_CrossSection(Material="Waveguide",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="Waveguide",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=5));
 };
layer("start",view=on,fill=off,width=73,refPlaneY=0,overlayfactor=1,Emk[60000,0,0],ColorRGB(205,255,0))
{
  XY_CrossSection(Material="start",structure,analyse,Mask=0,Line(Cx=0,Cw=$));
  XY_CrossSection(Material="start",structure,analyse,Rectangle(Cx=0,Cy=2.5,Cw=$,Ch=85));
 };

DefaultValue(Lambda=1.55,Width=65,ModeNum=0,Polarisation=TE,BpmOrder=2,viewmaskwindow=off,viewbpmwindow=on,viewmode="Normal",DiscretiseBpm=1e-020);
listcomment(Text="-----------------Definition of
Device------------------"); deflayer(layer "Waveguide");
variable(Name="nback",Value=1.5,Comment="");
variable(Name="ncore",Value=nback+0.027,Comment="");
variable(Name="nstart",Value=ncore,Comment="");
listcomment(Text="---------------input----------------");
variable(Name="inwidth",Value=0.75,Comment="");
listcomment(Text="---------------Stem----------------");
variable(Name="bigwidth",Value=25,Comment="");
listcomment(Text="------------Outputs--------------");
variable(Name="short",Value=3.35,Comment="");
variable(Name="mid",Value=4,Comment="");
variable(Name="Long",Value=4.8,Comment="");
variable(Name="dummy1",Value=7,Comment="");
variable(Name="dummy2",Value=(bigwidth-short-mid-dummy1-Long),Comment="");
listcomment(Text="----inoffset-------");
variable(Name="inoffset",Value=70,Comment="");
variable(Name="Ls",Value=50000,Comment="");
listcomment(Text="----output offset-------");
variable(Name="outoffset",Value=9,Comment="");
variable(Name="smode",Value=2,Comment="");
listcomment(Text="--------Radius Factor--------");
variable(Name="rad",Value=0.1,Comment="");
variable(Name="rad1",Value=0.1,Comment="");
puts(What="---------Beta Values---------");
```

```
variable(Name="beta0big",Value="modepropml('re','neff',0,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta1big",Value="modepropml('re','neff',1,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta2big",Value="modepropml('re','neff',2,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta3big",Value="modepropml('re','neff',3,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta4big",Value="modepropml('re','neff',4,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta5big",Value="modepropml('re','neff',5,5,nback,bigwidth,ncore,5,nback)",Comment="");
variable(Name="beta0short",Value="modepropml('re','neff',0,5,nback,short,ncore,5,nback)",Comment="");
variable(Name="beta1short",Value="modepropml('re','neff',1,5,nback,short,ncore,5,nback)",Comment="");
variable(Name="beta0mid",Value="modepropml('re','neff',0,5,nback,mid,ncore,5,nback)",Comment="");
variable(Name="beta1mid",Value="modepropml('re','neff',1,5,nback,mid,ncore,5,nback)",Comment="");
variable(Name="beta0Long",Value="modepropml('re','neff',0,5,nback,Long,ncore,5,nback)",Comment="");
variable(Name="beta1Long",Value="modepropml('re','neff',1,5,nback,Long,ncore,5,nback)",Comment="");
variable(Name="beta0dummy2",Value="modepropml('re','neff',0,5,nback,dummy2,ncore,5,nback)",Comment="");
variable(Name="beta1dummy2",Value="modepropml('re','neff',1,5,nback,dummy2,ncore,5,nback)",Comment="");
variable(Name="beta2dummy2",Value="modepropml('re','neff',2,5,nback,dummy2,ncore,5,nback)",Comment="");
puts(What="Bigstem");
variable(Name="beta2big",Value=iprint(beta2big),Comment="");
variable(Name="beta3big",Value=iprint(beta3big),Comment="");
variable(Name="beta4big",Value=iprint(beta4big),Comment="");
puts(What="Short");
variable(Name="beta0short",Value=iprint(beta0short),Comment="");
puts(What="Mid");
variable(Name="beta0mid",Value=iprint(beta0mid),Comment="");
puts(What="Long");
variable(Name="beta0Long",Value=iprint(beta0Long),Comment="");
puts(What="Dummy2");
variable(Name="beta1dummy2",Value=iprint(beta1dummy2),Comment="");
variable(Name="beta2dummy2",Value=iprint(beta2dummy2),Comment="");
listcomment(Text="----------WaveguideStructure--------");
taper(pin=1,pout=2,Start_Width=bigwidth,End_Width=bigwidth,Length=200,Offset=0,Angle_Out=0);
puts(What="-----------Outputs------------");
puts(What="----Width----"); puts(What="Dummy1"); SinBend(layer
"Waveguide",pin=2,pout=3,offset[0,bigwidth/2-dummy1/2,0],Width=iprint(dummy1),Length=Ls,Offset=2*outoffset,Sections=30);
puts(What="Short"); SinBend(layer
"Waveguide",pin=2,pout=4,offset[0,bigwidth/2-dummy1-short/2,0],Width=iprint(short),Length=Ls,Offset=outoffset,Sections=30);
puts(What="Mid");
taper(pin=2,pout=5,offset[0,bigwidth/2-dummy1-short-
mid/2,0],Start_Width=mid,End_Width=iprint(mid),Length=Ls,Offset=0,Angle_Out=0);
puts(What="Long"); SinBend(layer
"Waveguide",pin=2,pout=6,offset[0,bigwidth/2-dummy1-short-mid-Long/2,0],Width=iprint(Long),Length=Ls,Offset=-outoffset,Sections=30);
puts(What="Dummy2"); SinBend(layer
"Waveguide",pin=2,pout=7,offset[0,bigwidth/2-dummy1-short-mid-Long-dummy2/2,0],Width=iprint(dummy2),Length=Ls,Offset=-2*outoffset,Sections=30);

listcomment(Text="------------Input Source---------------");
fieldM(offset[1,0,0],Mode_No=smode,Range=bigwidth+2,Power=1,Phase=0,Tilt=0);

listcomment(Text="------------Mode Overlap---------------");
overlapM(pin=2,offset[-5,0,0],Label="bstem
1a",Mode_No=1,Range=bigwidth  +2,Tilt=0);
overlapM(pin=2,offset[-5,0,0],Label="bstem
2b",Mode_No=2,Range=bigwidth  +2,Tilt=0);
overlapM(pin=2,offset[-5,0,0],Label="bstem3c",Mode_No=3,Range=bigwidth
+2,Tilt=0);
overlapM(pin=2,offset[-5,0,0],Label="bstem4d",Mode_No=4,Range=bigwidth
+2,Tilt=0);

overlapM(pin=3,offset[-5,0,0],Label="dummy1
0a",Mode_No=0,Range=dummy1+2,Tilt=0);
overlapM(pin=3,offset[-5,0,0],Label="dummy1
1a",Mode_No=1,Range=dummy1+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="short
0a",Mode_No=0,Range=short+2,Tilt=0);
overlapM(pin=4,offset[-5,0,0],Label="short
1a",Mode_No=1,Range=short+2,Tilt=0);
overlapM(pin=5,offset[-5,0,0],Label="Mid 0a",Mode_No=0,Range=mid
+2,Tilt=0); overlapM(pin=5,offset[-5,0,0],Label="Mid
1a",Mode_No=1,Range=mid+2,Tilt=0);
overlapM(pin=6,offset[-5,0,0],Label="Long 0a",Mode_No=0,Range=Long
+2,Tilt=0); overlapM(pin=6,offset[-5,0,0],Label="Long
1a",Mode_No=1,Range=Long+2,Tilt=0);
overlapM(pin=7,offset[-5,0,0],Label="dummy2
0a",Mode_No=0,Range=dummy2+2,Tilt=0);
overlapM(pin=7,offset[-5,0,0],Label="dummy2
1a",Mode_No=1,Range=dummy2+2,Tilt=0);
```

# Bibliography


[1] "Characteristics of CWDM. Roots, current status and future opportunities", RBN inc 2002

[2] IEC's Florencio Canizales and Princetel's Barry Zhang, PhD"Back to Basics", UNDERWATER MAGAZINE,Article reprint - May/June 2003

[3] Teem Photonics, "Application Note: How to split signals in FTTH- PON Access Networks, (issue Date May 2003)"

[4] J. Hech, "Understanding Fibre Optics", Prentice Hall PTR (2002).

[5] "WDM Multiplexers, Demultiplexers and Filters" EE 233. Fall 2002, photonics.eecs.berkeley.edu/ee233f02/WDMux.ppt, 9/28/04

[6] A.Snyder and J.D Love, "Optical Waveguide Theory", Chapman and hall, 1983.

[7] J.D. Love and A. Ankiewicz Asymmetric Y-junction mode converter and wavelength de-multiplexer Proc. ACOFT, Sydney, Australia, 80-81 (2001)

[8] J.D. Love and A. Ankiewicz Purely geometrical coarse wavelength multiplexer/demultiplexer Electronics Letters, 39, 1384-5 (2003)

[9] A. Ankiewicz and J.D. Love Three-wavelength multiplexing/demultiplexing based on adiabatic mode evolution Proc. ACOLS, Melbourne, Australia, 167 (2003)

[10] RwcVance and J.d Love, Asymmetric adiabatic multiprong for mode -multiplexed systems

[11] J.A. Besley, J.D. Love and W. Langer "Mode de-multiplexing planar power splitter J. Lightwave Technology, 16, 678-684 (1998)

[12] J.D. Love "Application of a low-loss criterion to optical waveguides and devices" IEE Proceedings-J, Optoelectronics, 136, 225-228 (1989)

[13] Geshiro M, Kitamura T, Fukumura K, Sawa S,"A three-waveguide tapered-velocity coupler for dividing optical power into three equal parts",IEICE TRANSACTIONS ON ELECTRONICS E80C (11): 1414-1420 NOV 1997

[14] Geshiro M, Kitamura T, Yoshikawa T, Sawa S,"A two-waveguide tapered velocity coupler for a variable divider of optical power",IEICE TRANSACTIONS ON ELECTRONICS E79C (4): 587-592 APR 1996

[15] RAMASWAMY RV, KIM HS, “A NOVEL, TAPERED, BOTH IN DIMENSION AND IN INDEX, VELOCITY COUPLER”, IEEE PHOTONICS TECHNOLOGY LETTERS 5 (5): 557-560 MAY 1993

[16] Popovic M, Wada K, Akiyama S, Haus HA, Michel J, “Air trenches for sharp silica waveguide bends”,JOURNAL OF LIGHTWAVE TECHNOLOGY 20 (9): 1762-1772 SEP 2002

[17] Kovalev NF, “Adiabatic mode conversion in waveguides with shallow corrugations”,JOURNAL OF COMMUNICATIONS TECHNOLOGY AND ELECTRONICS 47 (8): 836-843 AUG 2002

[18] Kenis AM, Vorobeichik I, Orenstein M, Moiseyev N Source “Non-evanescent adiabatic directional coupler”,JOURNAL OF QUANTUM ELECTRONICS 37 (10): 1321-1328 OCT 2001

[19] F. Ladouceur and J.D. Love Silica-based Buried Channel Waveguides and Devices Chapman & Hall, London, 1996

[20] “Coupled Waveguide Modes and Mode Superposition”

`http://www.dur.ac.uk/g.h.cross/notes_f.pdf`