Career Path / Entry level Requirements

University is such a great place to learn some key fundamentals of your Engineering career, a lot of which may not be immediately apparent. I found the first year engineering fundamentals across a variety of disciplines has helped immensely when needing to collaborate with other teams and disciplines within the company. Your work is rarely an isolated project, and immediate stakeholders will likely be those internal to the company. Being able to have a base level of understanding allows you to understand their needs, and communicate effectively how your architecture or design can accomodate any problems they may encounter. Additionally, your ability to work on an engineering problem during group uni assignments and projects is invaluable. You need to learn flexibility in dealing with all different personality types, and solving interpersonal problems. You can put a bunch of smart people in a room together, but if you don't know how to position everyone so they can all play to their strengths, the groups effectiveness will be severely limited. I personally found the transition from student to professional quite easy, and this may be to the credit of Griffith's WIL program. I already had experience in the industry and company I transitioned to work for, and was able to gain this experience in a position where there was very little pressure and expectation of me. There was also a great level of support, because of my role as an intern at the time. This I believe helped me to ease into the professional role, and when I began work full time, I felt that I could focus 100% on my professional work as I no longer had to worry about the completion of my uni degree. In terms of the workload and complexity, Gilmour eased me into the responsibility - so it grew with the amount of experience I had. Within 6-12 months I was given my own projects, and within the first couple of years I became a Subject Matter Expert (SME) on the embedded flight software. I do feel the degree sufficiently prepared myself for a job in the industry. Working with others in the team also helped me to realise that even in industry, people are still constantly learning - so you don't need to be an expert straight out of uni to be valuable in the professional workforce.

Career progression at our company for an engineering can be within either the technical or people management stream, depending on your preference. There is a guideline for how long each progression should take between positions, however I'm glad that our company fosters the idea that advancement is largely based on merit - show to the company with tangible results that you are worthy of promotion and advancement. Our team is currently quite focused on quarterly KPIs which are guided by our head of department, but ultimately set for ourselves. Set ambitious KPIs and achieve them consistently, and you will progress very quickly. Performance reviews are done on a half yearly basis for position and remuneration negotiation.

Challenges / Jobs / Likes / dislikes

The biggest challenge is, due to the fact that it is safety critical real-time software, there is an exceptionally high level of quality required. This was overcome by continuous learning on embedded software best practice and leaning on mentors within the company who have had decades of experience within the aerospace industry, and constantly asking them for advice and feedback. Additionally, as the startup landscape is so dynamic, I have had to be very flexible in both my approach to writing software, and as an engineer - requirements are bound to change, you may have to do a bunch of rework, and some products/projects you have worked on for months may be scrapped. I believe if you can't deal with this, a startup environment is not for you. Software for the aerospace industry requires a higher standard of safety, reliability, and testing than many other industries due to the high stakes involved. Because of this, I believe it is the perfect place to learn fundamental skills and habits - I've since learnt software for other industries, and have found it to be very easy due to my previous learnings. I'm personally quite ambitious when it comes to handling challenging projects, so I will push myself to take on ambitious projects with aggressive schedules as a means to challenge myself and grow.

Skills

Technical skills are obviously very important, but the most undervalued skills are interpersonal and communication. You need to learn to sell your work, and work effectively within a team. I see so many super intelligent people with strong technical skills limit themselves because they lack the interpersonal skills. Technical: Work on some coding projects outside of work that interest you. Learn different languages, different projects that are outside of what you would usually work on: you'd be surprised at how much it can help your critical thinking and apply a different way of thinking to your main work. Interpersonal: Read books, watch youtube videos, go out and practice by meeting new people and making friends. Personality types: A range of personality types perform well in the industry. Strong technical and analytical skills are preferred for software, but you can find a role that suits you. There's a place for creative and emotional experience focused developers (ui/ux, front end design), and a place for super lowlevel technical (testing, implementation) and also more high level, abstract thinking (architecture, system design). It is preferred for graduates to demonstrate some talent through their own projects related to what role we are looking to fill, but mainly look for passion, enthusiasm, work ethic, and cultural fit. Technical skills can always be learnt, if you have the passion and work ethic. Additionally, regardless of how brilliant you may be, if you aren't a cultural fit, then you may not enjoy working with the team and may not feel like it is the right place for you. People work best when they enjoy what they do, and feel like they are making a difference.

Leadership / teamwork

Collaboration is essential in our workplace. We have regular team meetings and use project management tools to coordinate tasks. As mentioned before, great communication and interpersonal skills. A key skill is being able to communicate a complex idea to people that may have very little understanding of it - which is the case a lot of the time when collaborating with others in the team, and especially with other disciplines. As a mid-level engineer, I do have some leadership responsibilities. This involves mentoring junior engineers and coordinating team efforts on some of my projects or work that I am a subject matter expert for. Following standard software development practices, we have regular software reviews for any component of work that anyone in the team completes, and this is a great opportunity for coaching and mentoring. As mentioned before, I often seek guidance from some engineers in the team who have had decades of experience doing what I am currently doing. Even if the experience is not directly relevant, the way of thinking about the problem can be immensely helpful.

Future of industry / Retrospective / Schedule

The space industry in Australia is set to grow significantly over the next decade, with increased investment in sovereign launch capability, space exploration and satellite technology. Gaining skills early will be huge for future career prospects within the country. This will likely boost the Australian economy by creating jobs and inspiring innovation. Our company plans to continue developing cutting-edge technology and possibly exploring manned space missions. If I could go back, I would take more advantage of networking opportunities. Building a strong professional network can be immensely beneficial. My advice would be to maintain a balance between focusing on your studies and gaining practical experience through internships or projects. It may require some sacrifice to close off the final courses needed for the degree, but this one little push will all be worth it when you come to graduating and starting work full time. My daily duties involve code development, reviewing code, attending team meetings, testing software, and documenting software design/architecture. I plan these based on the company, department and team priorities and deadlines. Know when you need to surge and invest the extra hours, and when you need to focus on your personal life. You can't give it your all if you end up burning out. However, there will be times when you may need to sacrifice some days to ensuring you can deliver on ambitious goals you have set for your own progression and growth,.